JUST-NLP 2025

The 1st Workshop on NLP for Empowering Justice

Proceedings of the Workshop

December 24, 2025

Order copies of this and other ACL proceedings from:

Association for Computational Linguistics (ACL)
317 Sidney Baker St. S
Suite 400 - 134
Kerrville, TX 78028
USA
Tel: +1-855-225-1962
acl@aclweb.org

ISBN 979-8-89176-312-8

Introduction

We are pleased to share the Proceedings of JUST-NLP 2025: The First Workshop on NLP for Empowering Justice, which was held concurrently with IJCNLP-AACL 2025. This workshop represents a significant step forward in strengthening the connection between Natural Language Processing (NLP) and the legal field. This is especially true in places like India, where numerous languages are spoken, numerous cases need to be resolved, and access to legal services is limited. Researchers, practitioners, and technologists who work at the cutting edge of Legal-NLP were supposed to meet at JUST-NLP. Legal NLP is an area that requires methodological rigor, domain sensitivity, and careful consideration of real-world, high-stakes applications.

Obtaining legal information remains a challenge worldwide. In the Indian judiciary, particularly in lengthy case decisions, the use of complex legal reasoning structures and the fact that English is the primary language used in official proceedings make it challenging to ensure transparency and fairness. Automating key aspects of legal information processing, such as summarization, translation, retrieval, and reasoning, could make tasks easier for lawyers and citizens. The primary goal of JUST-NLP 2025 was to accelerate progress in these areas and establish a robust research community centered on Legal-NLP.

Overall, our first edition featured a research track, four invited talks, and two shared tasks: (1) L-SUMM, an abstractive summarization task for Indian legal judgments, and (2) L-MT, a legal machine translation task between English and Hindi. The introduction of two shared tasks, each addressing important needs in legal information processing, was a highlight of this year's workshop. The workshop received 29 submissions, of which 21 were accepted, highlighting a strong community interest in Legal NLP and its applications to the legal processing pipeline. Among the accepted papers, 5 were regular research-track papers published in the proceedings, and 2 were accepted as non-archival presentations. For the shared tasks, 9 papers were accepted for L-SUMM, and 5 papers were accepted for L-MT, for publication in the proceedings. Since the event was held in a hybrid format, presentations were delivered both in person at the IJCNLP-AACL 2025 venue in Mumbai and virtually.

The Shared Task on Summarization of Indian Court Judgments (L-SUMM) addressed the challenge of creating concise, clear, and accurate summaries of lengthy legal texts. It added the InLSum dataset, which has 1,800 court decisions and expert-written, abstract summaries of those decisions. Nine teams that participated examined a wide range of methods, including long-context transformers, hierarchical and multi-stage summarization frameworks, extractive-abstractive hybrids, rhetorical-role-aware chunking, and reinforcement-learning-based alignment. The results demonstrate the importance of long-context modeling and adapting to specific fields in creating effective legal summaries.

The Shared Task on English-to-Hindi Legal Machine Translation (L-MT) also aimed to make India's multilingual legal system more accessible. Using a carefully chosen 50,000-sentence English–Hindi legal parallel corpus and a thorough evaluation framework, the participating teams tested multilingual Transformers, QLoRA-based parameter-efficient fine-tuning, curriculum learning, reinforcement learning with verifiable rewards, and even Transformer models trained from scratch. The results show that domain-adaptive finetuning and precision-focused optimization are effective for getting accurate legal translations. This shows that MT systems can be useful in real-world legal workflows.

In addition to the shared tasks, the workshop included research contributions on understanding legal texts, reasoning with statutes and case law, extracting information from specific domains, mining arguments, adapting legal-domain LLMs, and using AI for justice in a socially responsible way. The papers in this volume demonstrate that this research area is maturing, but they also highlight some problems that still need to be addressed, such as ensuring long-form generation is factually consistent, reducing hallucinations in high-stakes fields, and incorporating expert-driven evaluation protocols.

Our workshop also featured a diverse set of invited talks from leading experts across law, industry, public institutions, and AI research. The talks highlighted multiple perspectives on the future of Legal NLP, ranging from legal theory and courtroom-scale AI deployment to ontology-driven knowledge representations and technology-enabled governance reform.

We believe that the resources, ideas, and discussions emerging from JUST-NLP 2025 will set the stage for future progress at the intersection of law and language technologies. As NLP systems improve, they are more likely to make legal knowledge accessible to everyone, making it easier for all to access justice. We hope that this workshop will foster greater collaboration, responsible innovation, and a deeper understanding of the unique challenges that Legal-NLP presents.

We would like to thank the authors, shared-task organizers, program committee members, reviewers, and participants for their hard work and assistance. We would also like to thank the individuals who helped organize the IJCNLP–AACL 2025 event. We are excited to see JUST-NLP continue to grow in the years to come.

Organizing Committee

Program Chairs

Ashutosh Modi, IIT Kanpur, India
Saptarshi Ghosh, IIT Kharagpur, India
Asif Ekbal, IIT Patna, India
Pawan Goyal, IIT Kharagpur, India
Sarika Jain, NIT Kurukshetra, India

Web and Publicity Chairs

Abhinav Joshi, IIT Kanpur, India
Shivani Mishra, IIT Kanpur, India

Shared Task and Publication Chairs

Debtanu Datta, IIT Kharagpur, India
Shounak Paul, IIT Kharagpur, India
Kshetrimayum Boynao Singh, IIT Patna, India
Sandeep Kumar, IIT Patna, India

Program Committee

Program Committee

Abhinav Joshi, IIT Kanpur, India
Abhisek Sharma, NIT Kurukshetra, India
Amita Jain, NSUT Delhi, India
Aniket Deroy, IIT Delhi, India
Debtanu Datta, IIT Kharagpur, India
Kshetrimayum Boynao Singh, IIT Patna, India
Pooja Harde, NIT Kurukshetra, India
Sandeep Kumar, IIT Patna, India
Shivani Mishra, IIT Kanpur, India
Shounak Paul, IIT Kharagpur, India
Sumit Dalal, Bennett University, India

Invited Speakers

Niraj Kumar, NLU Delhi, India
Arghya Bhattacharya, Adalat AI, India
Joseph Pookkatt, Staram, India
Sampritha Manjunath, Insight SFI Centre for Data Analytics, Ireland
Parth Parikh, eSuccess AI Technologies, India
Nishi Yadav, Ministry of Tribal Affairs, Government of India, India

Keynote Talk

Artificiality of Law in reference to Artificial Intelligence

Niraj Kumar
NLU Delhi, India

Abstract: This talk examines the evolving relationship between legal theory and emerging AI systems, drawing on insights from constitutional and comparative law. It explores how AI challenges traditional conceptions of legal reasoning, authority, and legitimacy, and raises foundational questions about the artificiality of law itself. The talk also reflects on the implications of entrusting interpretive and decision-support roles to artificial agents within legal systems.

Bio: Prof. Niraj Kumar is a Professor of Law and Director of the Centre for Comparative Law at National Law University Delhi, India. His areas of expertise include Constitutional Law, Administrative Law, Comparative Law, Environmental Law, and Legal Theory. He previously served as Additional Registrar (Research) at the Supreme Court of India, attached with Hon'ble the Chief Justice of India from January 2019 to May 2021. He has authored several notable books, including The Indian Legal System: An Enquiry (Oxford, 2019) and multiple volumes of the Indian Yearbook of Comparative Law (Springer). His research has also been published in leading journals, including the Journal of the Indian Law Institute, NUJS Law Review, and Shimla Law Review. He has also been a resource person for training officers from the IAS, IPS, Judicial Services, and other civil services.

Keynote Talk

Building AI for India's Courtrooms: Scaling Sovereign Voice and Language Systems Across 20 percent of the Judiciary

Arghya Bhattacharya
Adalat AI, India

Abstract: India is home to the world's largest and most linguistically complex justice system – one that processes millions of hearings every day across dozens of languages and procedural contexts. Over the last two years, Adalat AI has deployed voice and language AI in nearly 20% of all Indian courtrooms, powering live transcription, dictation, translation, and document navigation at a population scale. This talk walks through the journey of building these systems end-to-end: how we approached the machine-learning stack for courts, how we handle legal context across diverse tasks, and what it takes to engineer sovereign, privacy-preserving AI for public institutions. I'll share the technical and operational challenges we encountered, speech variability, multilingual legal phrasing, courtroom acoustics, edge-case reasoning, and deployment constraints, and the architectural decisions that allowed us to scale from pilots to thousands of courtrooms. The session will highlight lessons on building domain-grounded AI for high-stakes environments, discuss why voice AI is only one part of a much broader AI for Justicestack, and outline what it means to build reliable, inclusive, and future-proof digital public goods for the Global South.

Bio: Mr. Arghya Bhattacharya is the Co-Founder and CTO of Adalat AI, where he leads the development of sovereign voice and language technologies that now power courtrooms across India. A researcher and engineer by training, he holds a Bachelor's in Computer Science and a Master's in Artificial Intelligence from IIIT Hyderabad, with publications in leading venues including ACL, EMNLP, CoNLL, and EaMT. Before launching Adalat AI, Arghya was the first founding engineer at Enterpret, where he built large-scale NLP systems for customer intelligence, and later worked at Equal on identity-verification infrastructure for high-security environments. His academic work in multilingual and low-resource NLP underpins Adalat AI's vernacular court-workflow stack. In 2024, he co-founded Adalat AI with Utkarsh Saxena to modernize India's justice system through AI. Today, their platform supports live transcription, dictation, translation, and legal analytics across thousands of courts. In 2025, Arghya was named to the Forbes 30 Under 30 Asia – Social Impact list for advancing accessible, rights-aligned AI for public institutions. His broader mission is to build sovereign, privacy-preserving digital public goods that enhance inclusion, strengthen state capacity, and support justice systems across the Global South.

Keynote Talk

Designing and Implementing Knowledge Graphs in the Legal Domain

Joseph Pookkatt, Sampritha Manjunath, Parth Parikh
Staram, India; Insight SFI Centre for Data Analytics, Ireland; eSuccess AI Technologies, India

Abstract: Ontologies play a crucial role in enabling data integration from multiple sources by providing a common vocabulary for diverse systems. This is particularly important for AI applications, which often rely on data from various databases and formats. Ontologies align this disparate data, making seamless processing possible. They also empower AI to perform semantic searches, interpreting user intent rather than just matching keywords. This allows users to make more natural language queries, enabling AI to retrieve precise and meaningful information. In the realm of Retrieval-Augmented Generation (RAG), the dominant approach has been BaselineRAG. However, BaselineRAG has a notable limitation: it struggles to connect the dots between concepts that are semantically distant but logically connected. This is where Knowledge Graphs (KGs) excel. By exploring relationships within the graph using graph embeddings or graph-specific algorithms, GraphRAG captures the context and logical relationships between concepts. This makes it far better positioned to deliver higher-quality, more accurate outputs compared to traditional RAG methods. In the context of Indian law, building a large, unified knowledge graph has historically been too challenging due to the heterogeneous systems involved. Staram is tackling this issue by customizing multiple knowledge graphs for specific legal use cases in India. They will demonstrate how Large Language Models (LLMs) can infer graph architectures and symbolic layers tailored to specific needs. This approach enables the creation of smaller, specialized graphs that are accurate, interoperable, and easier to govern. These advancements hold significant promise for building generative AI applications that transform legal research, compliance, and decision-making.

Bio: 1. Joseph Pookkatt: Mr. Joseph Pookkatt is a practicing lawyer with over 30 years of experience at the intersection of corporate law, litigation, legal technology, AI, and entrepreneurship. As a co-founder of Staram Analytics, Joseph collaborates with legal policy organizations to develop ontologies for specific laws and is leading the development of India's first open-source legal knowledge graph. His work aims to make Indian legal systems more accessible, efficient, and transparent through the power of AI and ontologies.
2. Sampritha Manjunath: Ms. Sampritha Manjunath is an NLP Data Scientist with 9+ years of experience in language models, knowledge graphs, RAG, and conversational AI. Experienced in leading research projects, developing scalable AI solutions, and publishing in top venues such as ACL and LDK. Currently a Research Associate at the University of Galway, focused on building impactful and responsible AI systems.
3. Parth Parikh: Mr. Parth Parikh is an AI Research Engineer at eSuccess AI Technologies, where he leads work in legal NLP, knowledge graphs, and ontology-driven AI for large-scale legal reasoning. He earned his M.Tech in Computer Science from IIT Bombay (2023), with research focused on applied machine learning and Computer Vision. At eSuccess AI, Parth works on building advanced legal intelligence systems powered by LLMs and retrieval-augmented architectures. His current interests include structured legal representations, legal data models, and domain-adapted language models.

Keynote Talk

Fixing the Process, Powering the System: What Jharkhand Teaches India About Legal-Tech

Nishi Yadav
Ministry of Tribal Affairs, Government of India, India

Abstract: Access to justice is a core democratic promise, yet its delivery in India remains deeply strained. With 4.7 crore cases pending across courts, delays routinely turn justice delayedinto justice denied.The Government, Centre and States combined, is the single largest litigant, responsible for nearly half of all cases, including 46% of High Court matters, many of which arise from service-related disputes in departments like Education, Railways, Health, and Finance. This volume of litigation imposes severe costs: rising legal expenditure, reduced administrative bandwidth, and declining institutional trust. In several departments, administrators spend over 40% of their time managing court matters. Right from 1974 by Justice Krishna Iyer, the 100th Law Commission Report - 1983, the 126th Law Commission Report – 1988, to the Prime Minister leading meetings between the Department of Justice with the Central Government, and the more recent attempt to accelerate a National Litigation Policy, the issue has been acknowledged and attempts have been made at reform. A consistent theme in all of these efforts has been that of wanting to reduce litigation involving the Govt. Experience across states shows that government litigation is rarely only a legal problem. It stems from incoherent policies, weak grievance redressal, and limited institutional capacity for managing legal processes. Inconsistent decisions, poor tracking of judicial precedents, and the absence of mechanisms to resolve disputes internally often push routine administrative issues into courts. This systemic challenge sets the stage for why Jharkhand's legal-tech transformation is particularly notable. The Case of Jharkhand: While digital governance conversations often highlight Telangana, Karnataka, or Kerala, one of India's most quietly impactful reforms has unfolded in Jharkhand - a young, resource-rich, predominantly tribal state. Its legal-tech journey stands out for prioritising process redesign before technology adoption. Jharkhand followed a pragmatic approach: map processes, fix bottlenecks, then digitise. This ensured that technology supported well-defined workflows instead of automating inefficiencies. The state's journey began with the Vidhi Portal (2016) in the Advocate General's Office, which evolved into the central platform for real-time service of notices and orders. Building on this foundation, the Integrated Litigation Management System (ILMS) launched in 2020 by the Department of School Education & Literacy made the department genuinely data-driven. Case pendency dropped sharply; response times shrank from 45–70 days to just 7 days; and administrative culture shifted from reactive firefighting to proactive governance. This cooperative ecosystem, spanning the AG Office, the Department of IT & e-Governance, and key line departments, stands in contrast to national systems like LIMBS, which function largely as data-entry repositories with limited integration or decision-support capability. Jharkhand is now preparing its next leap: AI-enabled predictive governance to identify recurring disputes, strengthen policy design, and reduce litigation at its root. This talk positions Jharkhand's experience not just as a state-level success but as a scalable blueprint for India, where governance reform begins with process, matures with technology, and evolves with intelligence.

Bio: An Economics graduate from DU and a lawyer with over 11 years of PQE, Ms. Nishi began her career engaging with underserved communities through organizations like WGHR-UN, NLUD, and HRLN. Her tenure at the Chief Minister's Office, Delhi Government, ignited her passion for understanding and improving government systems. Since 2018, she has been working closely with the Government of Jharkhand and various non-profit organisations, leading transformative initiatives to drive policy reforms and strengthen governance, developing expertise in analyzing legal processes from constitutional, technological, and organizational perspectives to improve efficiency and effectiveness. She spearheaded the study, design, development, and implementation of the Integrated Litigation Management System (ILMS) with the Government of Jharkhand which automated file movement processes, reducing the pendency of court cases from 11,000 to 3,500. Alongside, she improved key processes such as leave management, teacher transfers, and school monitoring, integrating legal-tech solutions with system-wide reforms. Additionally, she has conducted extensive studies in close association with bureaucracy in other states, combining legal acumen, technological insights, and strategic vision. Committed to socio-economic and governance reforms, her collaborations with various state governments highlight her ability to navigate the intricacies of public systems and officers while building systems that drive meaningful impact. She is currently involved in strategic governance and legal-tech projects with the Ministry of Tribal Affairs (GoI), the Government of Jharkhand in collaboration with IIT-Kanpur and IIT (ISM) Dhanbad, and ongoing reform initiatives with the Government of Jammu & Kashmir.

Table of Contents

Overview of the 1st Workshop on NLP for Empowering Justice

Ashutosh Modi[1], Saptarshi Ghosh[2], Asif Ekbal[3], Pawan Goyal[2], Sarika Jain[4], Abhinav Joshi[1], Shivani Mishra[1], Debtanu Datta[2], Shounak Paul[2], Kshetrimayum Boynao Singh[3], Sandeep Kumar[3]

[1]IIT Kanpur, India, [2]IIT Kharagpur, India, [3]IIT Patna, India, [4]NIT Kurukshetra, India

Correspondence: ashutoshm@cse.iitk.ac.in

Abstract

The first iteration of the JUST-NLP: Workshop on NLP for Empowering Justice was organized to accelerate research in Natural Language Processing for legal text processing. The inaugural edition, JUST-NLP 2025, was held as a hybrid event at IJCNLP-AACL 2025 on December 24 at IIT Bombay. The program featured a research track, four invited talks, and two shared tasks: (1) L-SUMM, an abstractive summarization task for Indian legal judgments, and (2) L-MT, a legal machine translation task between English and Hindi. The workshop received strong interest from the community, with 29 submissions, of which 21 were accepted. Among the accepted papers, 5 were regular research-track papers published in the proceedings, and 2 were accepted as non-archival presentations. For the shared tasks, 9 papers were accepted for L-SUMM, and 5 papers were accepted for L-MT, for publication in the proceedings. The workshop focused on a broad set of Legal NLP challenges, including information extraction, retrieval, multilingual processing, legal reasoning, and applications of large language models. Overall, JUST-NLP 2025 aimed to bring together AI researchers and legal practitioners to develop scalable, domain-aware NLP methods that can support legal workflows and contribute toward more efficient and equitable justice systems.

1 Introduction

Legal documents form the textual backbone through which societies articulate rights, enforce responsibilities, and administer justice. They encode statutory frameworks, capture the reasoning behind judicial decisions, and document complex interactions among individuals, institutions, and the state. As a result, Legal NLP has emerged in recent years as a significant research area: legal texts are not only primary to ensuring fairness, transparency, and accountability in democratic systems, but also indispensable for facilitating faster and more structured judicial processes. However, these documents pose challenges that are far more demanding than those encountered in many other NLP domains, owing to their specialized language, intricate reasoning structures, and strong reliance on precedent and contextual interpretation.

Legal documents often require a precise and unambiguous understanding, a high level of factual and logical consistency, and substantial background/contextual knowledge for processing. Moreover, these documents span hundreds of pages, follow rigid structural conventions, and contain dense argumentation supported by layered citations to statutes, procedural rules, and prior cases. In multilingual jurisdictions such as India, these challenges are even greater, where legal information is spread across multiple languages, various court levels, and diverse document types, including judgments, petitions, orders, and statutes, each with its own linguistic and semantic characteristics. Such complexity motivates the need for domain-specific models, datasets, and evaluation protocols, as general-purpose NLP methods frequently fail to capture these nuances.

Countries with large populations, most notably India, with over 44 million pending cases, struggle with delays driven by manual workflows, difficulty in locating relevant precedents, and limited accessibility of legal information for both practitioners and citizens. Advances in NLP and large language models provide a promising avenue to mitigate these bottlenecks through tasks such as automated summarization, translation, precedent retrieval, and assistive legal reasoning. However, progress has been slowed by the scarcity of annotated legal datasets, the limited collaboration between technologists and legal experts, and the particularly high stakes of deploying AI in legal settings.

The JUST-NLP workshop was created to address these challenges by bringing together researchers from NLP, information retrieval, machine learn-

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 1–4

ing, and AI ethics, alongside law practitioners and scholars. Its goal is to provide a dedicated venue for the development of domain-aware models, high-quality legal datasets, multilingual resources, and discussions on the risks and opportunities of AI for justice. JUST-NLP invited research contributions spanning legal reasoning, information extraction, statute and precedent retrieval, multilingual legal processing, and applications of large language models in legal workflows. Overall, the program featured a research track, four invited talks (detailed in Section 3), and two shared tasks: L-SUMM and L-MT (detailed in Section 4).

As the first edition of the workshop co-located with IJCNLP-AACL 2025, JUST-NLP aims to establish a sustained, interdisciplinary forum that promotes both the scientific advancement of Legal NLP and its responsible deployment toward more efficient, inclusive, and equitable justice systems.

2 Program

The first iteration of the JUST-NLP 2025 workshop featured a research track with open submissions, four invited talks, and two shared task tracks. The workshop received 29 submissions, of which 21 were accepted, highlighting a strong community interest in Legal NLP and its applications to the legal processing pipeline. Among the accepted papers, 5 were regular research-track papers published in the proceedings, and 2 were accepted as non-archival presentations. For the shared tasks, 9 papers were accepted for L-SUMM, and 5 papers were accepted for L-MT, for publication in the proceedings. Since the event was held in a hybrid format, presentations were delivered both in person at the IJCNLP-AACL 2025 venue in Mumbai and virtually.

The accepted papers provided a broad spectrum of Legal NLP research, highlighting community interest in both foundational challenges and emerging directions in the field of legal text processing. The contributions/submissions aim to address legal information extraction, knowledge graph construction, statute and precedent retrieval, legal citation network modeling, consumer-law assistance systems, and specialized pipelines for legal analytics. A significant portion of the accepted work focused on multilingual and English–Hindi legal machine translation, as well as a wide range of approaches to legal document summarization, including hierarchical and structure-aware chunking, retrieval-augmented, and agentic LLM workflows. Across these efforts, a clear trend emerged as the integration of large language models into legal tasks, with a particular emphasis on long-document processing, domain adaptation, reinforcement learning–based fine–tuning, and hybrid extractive–generative strategies. Overall, these papers highlight the community's growing interest in scalable, transparent, and domain-grounded LLM methods for real-world legal reasoning and decision support.

In addition to the research track, the workshop also hosted two shared tasks, L-SUMM, an abstractive summarization task for Indian legal judgments, and L-MT, a legal machine translation task between English and Hindi. Both tasks demonstrated active participation from multiple research groups, highlighting the growing interest in this area.

3 Invited Talks

The JUST-NLP 2025 workshop featured a diverse set of invited talks from leading experts across law, industry, public institutions, and AI research. The talks highlighted multiple perspectives on the future of Legal NLP, ranging from legal theory and courtroom-scale AI deployment to ontology-driven knowledge representations and technology-enabled governance reform.

3.1 Prof. Niraj Kumar

Affiliation: National Law University Delhi
Title: Artificiality of Law in Reference to Artificial Intelligence
Summary: Prof. Niraj Kumar delivered a keynote examining the evolving relationship between legal theory and emerging AI systems. Drawing on his expertise in constitutional and comparative law, he reflected on how AI challenges traditional conceptions of legal reasoning, authority, and legitimacy. His talk explored foundational questions around the "artificiality" of law itself and the implications of entrusting interpretive or decision-support roles to artificial agents.

3.2 Mr. Arghya Bhattacharya

Affiliation: Co-Founder & CTO, Adalat AI
Title: Building AI for India's Courtrooms: Scaling Sovereign Voice & Language Systems Across 20% of the Judiciary
Summary: Mr. Arghya Bhattacharya shared the technical and operational journey behind deploying large-scale voice and language AI across thousands of Indian courtrooms. His talk detailed the

engineering of sovereign, privacy-preserving systems for live transcription, dictation, translation, and legal assistance, addressing challenges such as courtroom acoustics, multilingual phrasing, domain grounding, and reliable scaling in high-stakes public institutions. The session also outlined how voice AI forms part of a broader "AI for Justice" ecosystem and discussed pathways for building inclusive digital public goods for the Global South.

3.3 Mr. Joseph Pookkatt, Ms. Sampritha Manjunath, and Mr. Parth Parikh

Affiliation: Staram Analytics & eSuccess AI Technologies
Title: Designing and Implementing Knowledge Graphs in the Legal Domain
Summary: This invited industry panel examined the role of ontologies and knowledge graphs in enabling robust, interpretable, and domain-grounded AI systems for legal applications. The speakers highlighted why traditional RAG pipelines struggle with semantically distant but logically related concepts and demonstrated how GraphRAG and structured symbolic layers can significantly improve retrieval and reasoning. Using Indian legal use cases, they showcased approaches for building smaller, specialized legal knowledge graphs and discussed how LLMs can assist in graph architecture inference, supporting next-generation legal research, compliance, and decision-making systems.

3.4 Ms. Nishi Yadav

Affiliation: Senior Legal Consultant, Ministry of Tribal Affairs, Government of India
Title: Fixing the Process, Powering the System: What Jharkhand Teaches India About Legal-Tech
Summary: Ms. Nishi Yadav presented a governance-focused perspective on legal-tech transformation, using Jharkhand's pioneering reforms as a case study. Her talk argued that sustainable legal-tech begins with process redesign, mapping institutional workflows, fixing structural bottlenecks, and only then introducing technology. The session traced Jharkhand's evolution from the Vidhi Portal to the Integrated Litigation Management System (ILMS), highlighting sharp reductions in case pendency, faster response cycles, and cultural shifts toward data-driven governance. She also outlined the state's next phase: integrating AI-enabled predictive governance to identify systemic drivers of litigation and strengthen policy design.

4 Shared Tasks

JUST-NLP 2025 hosted two shared tasks, each organized by members of the workshop organizing committee. Detailed descriptions, datasets, baselines, and system analyses are provided in the individual shared task overview papers (Datta et al., 2025; Singh et al., 2025) included in the proceedings.

Legal Summarization (L-SUMM) The L-SUMM (Datta et al., 2025) task focused on abstractive summarization of Indian legal judgments. Participants were required to generate concise and coherent summaries that capture the core legal reasoning and outcomes of complex, lengthy judicial documents. The task highlighted key challenges in legal summarization, including domain-specific terminology, multi-paragraph argumentation structures, and the need for faithful condensation of legal rationale. Evaluation was conducted using ROUGE-2, ROUGE-L, and BLEU, and participating teams explored a variety of approaches, including domain-adapted LLMs and long-context transformers. Full details and results appear in the corresponding task overview paper (Datta et al., 2025).

Legal Machine Translation (L-MT) The L-MT (Singh et al., 2025) task targeted English–Hindi legal machine translation, addressing the need for bilingual accessibility in India's multilingual judicial system. The task required systems to handle complex legal syntax and terminology while preserving semantic and legal fidelity. Submissions were evaluated using BLEU, METEOR, TER, chrF++, BERTScore, and COMET. Approaches ranged from fine-tuned encoder–decoder models to instruction-tuned large language models. A separate overview paper (Singh et al., 2025) in this volume provides a detailed presentation of the dataset, baselines, and performance analysis.

5 Workshop Overview and Outlook

The organizers were encouraged by the strong and diverse response to the inaugural JUST-NLP workshop. Accepted contributions spanned a wide spectrum, from applied systems addressing legal document summarization, translation, and knowledge retrieval, to foundational research exploring legal reasoning, multilingual NLP, and domain-adapted large language models. This breadth demonstrates both the societal relevance of NLP for justice and the technical richness of the domain.

JUST-NLP has helped bring together an emerging research community focused on legal NLP, broad enough to foster interdisciplinary collaboration, yet focused enough to make rapid progress on high-impact challenges. The workshop sits at the intersection of NLP, information retrieval, AI for governance, and legal studies, reflecting the need for cross-disciplinary approaches to improve access to justice.

Looking ahead, the workshop aims to continue expanding both its technical and thematic scope. Future editions will encourage contributions on fairness, explainability, and ethical AI in legal systems, as well as multilingual, cross-jurisdictional, and low-resource challenges. Building on the success of the shared tasks in legal summarization and machine translation, future iterations will continue to provide concrete problem settings that engage the research community and address pressing societal needs. The organizers hope that JUST-NLP will grow into a sustained forum for innovation, collaboration, and responsible deployment of NLP technologies in support of equitable and efficient justice systems.

6 Conclusion

Legal documents present a rich and challenging domain for NLP research, with high societal impact and pressing real-world applications. While prior work in information retrieval, knowledge representation, and computational linguistics provides a foundation, many critical challenges remain, ranging from understanding complex reasoning in judgments to handling multilingual and low-resource legal contexts. The automated processing of legal texts is still in its early stages, and careful, domain-aware approaches are essential to ensure accuracy, fairness, and reliability.

By providing a dedicated forum for interdisciplinary collaboration, the inaugural JUST-NLP workshop has highlighted these challenges and offered opportunities for researchers and practitioners to exchange ideas, benchmark systems, and explore innovative solutions. We hope that future editions of JUST-NLP will continue to foster the development of domain-specific models, curated datasets, and practical applications, ultimately contributing to a more efficient, accessible, and equitable justice system in India and beyond.

References

Debtanu Datta, Shounak Paul, Kshetrimayum Boynao Singh, Sandeep Kumar, Abhinav Joshi, Shivani Mishra, Sarika Jain, Asif Ekbal, Pawan Goyal, Ashutosh Modi, and Saptarshi Ghosh. 2025. Findings of the JUST-NLP 2025 Shared Task on Summarization of Indian Court Judgments. In *Proceedings of the First Workshop on NLP for Empowering Justice (JUST-NLP)*, Mumbai, India. Association for Computational Linguistics.

Kshetrimayum Boynao Singh, Sandeep Kumar, Debtanu Datta, Abhinav Joshi, Shivani Mishra, Shounak Paul, Pawan Goyal, Sarika Jain, Saptarshi Ghosh, Ashutosh Modi, and Asif Ekbal. 2025. Findings of the JUST-NLP 2025: A Shared Task on English-to-Hindi Legal Machine Translation. In *Proceedings of the First Workshop on NLP for Empowering Justice (JUST-NLP)*, Mumbai, India. Association for Computational Linguistics.

Findings of the JUST-NLP 2025 Shared Task on Summarization of Indian Court Judgments

Debtanu Datta[1], Shounak Paul[1], Kshetrimayum Boynao Singh[2], Sandeep Kumar[2], Abhinav Joshi[3], Shivani Mishra[3], Sarika Jain[4], Asif Ekbal[2], Pawan Goyal[1], Ashutosh Modi[3], Saptarshi Ghosh[1]

[1]IIT Kharagpur, India, [2]IIT Patna, India, [3]IIT Kanpur, India, [4]NIT Kurukshetra, India

Correspondence: debtanudatta04@gmail.com

Abstract

This paper presents an overview of the *Shared Task on Summarization of Indian Court Judgments* (L-SUMM), hosted by the JUST-NLP 2025 Workshop at IJCNLP-AACL 2025. This task aims to increase research interest in automatic summarization techniques for lengthy and intricate legal documents from the Indian judiciary. It particularly addresses court judgments that contain dense legal reasoning and semantic roles that must be preserved in summaries. As part of this shared task, we introduce the **In**dian Legal **Sum**marization (**InLSum**) dataset, comprising 1,800 Indian court judgments paired with expert-written abstractive summaries, both in English. Therefore, the task focuses on generating high-quality abstractive summaries of court judgments in English. A total of 9 teams participated in this task, exploring a diverse range of methodologies, including transformer-based models, extractive-abstractive hybrids, graph-based ranking approaches, long-context LLMs, and rhetorical-role-based techniques. This paper describes the task setup, dataset, evaluation framework, and our findings. We report the results and highlight key trends across participant approaches, including the effectiveness of hybrid pipelines and challenges in handling extreme sequence lengths.

1 Introduction

Legal case judgments are often lengthy, intricate, and densely packed with domain-specific terminology and complex judicial reasoning. As a result, legal professionals must manually review extensive case documents to identify relevant precedents, which is crucial in Common Law systems, such as in the Indian Judiciary (Bhattacharya et al., 2019). This is a process that is not only time-consuming but also cognitively demanding domain-expert knowledge. Prior research highlights that case judgments are substantially longer than documents in most other domains, and expert-written summaries are very expensive to obtain, leading to limited availability of high-quality supervised data for the legal summarization task (Shukla et al., 2022; Datta et al., 2023). Therefore, automatic summarization of case judgments has emerged as a crucial task in the Legal-NLP field aimed at reducing the burden on law practitioners and improving access to essential case information (Zhong et al., 2019; Datta et al., 2023). Recent developments in transformer-based abstractive models, including legal-domain adaptations in Large Language Models (LLMs), have further accelerated interest in building reliable, faithful, and coherent summaries of judicial rulings (Shukla et al., 2022; Sharma et al., 2023; Deroy et al., 2025).

Motivated by these challenges, the Shared Task on Legal Summarization (L-SUMM)[1] has been organized as part of the JUST-NLP 2025 Workshop, co-located with IJCNLP–AACL 2025. The primary aim of this task is to foster increased research interest in Legal-NLP, particularly in summarization methodologies for lengthy judicial texts, reflecting real-world requirements faced by legal information retrieval systems and practitioners.

In this regard, we introduce **InLSum** (**In**dian Legal **Sum**marization), a dataset comprising 1,800 case judgments from prominent Indian courts, along with expert-written abstractive summaries, both in English. Participants were required to produce abstractive summaries that capture the core reasoning and factual elements of these court rulings under a unified evaluation framework. Beyond benchmarking, this shared task offers valuable insights into the practical challenges of legal summarization, including handling extreme input lengths, ensuring factual consistency, and maintaining coherence across lengthy narrative structures.

A total of 9 teams participated in the L-SUMM

[1]https://exploration-lab.github.io/JUST-NLP/task/

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 5–11
December 24, 2025

InLSum dataset	Train		Validation		Test	
	Judgment	**Summary**	**Judgment**	**Summary**	**Judgment**	**Summary**
Number of Samples	1200	1200	200	200	400	400
Average Number of Words	8294	606	6815	603	7920	621
Mean Compression Ratio	1:14		1:12		1:12	

Table 1: Dataset Statistics of **InLSum** dataset.

Shared Task. In this paper, we provide an overview of the task in §2, including details of the dataset and evaluation metrics. We present the approaches adopted by the participating teams in §3 and §4 presents the results and discusses key insights. Finally, §5 concludes the paper.

2 Task Description

The L-SUMM Shared Task focuses on generating concise and coherent abstractive summaries of Indian court judgments in English, addressing the lengthy and intricate nature of these documents. Participants were provided with the **InLSum** dataset, which has dedicated train, test, and validation splits for developing models and tailoring them to legal summarization. Detailed description of the dataset is provided in §2.1. Participants used the provided *train* split to train and fine-tune their language models. During the Training Phase, they were required to submit predictions on the *validation* split to evaluate model performance and perform necessary tuning. In the Testing Phase, participants submitted predictions on the held-out *test* split to assess their models on unseen data. The final evaluation was conducted on this *test* set (the test data was released only after the completion of the Training Phase). The shared task was hosted on the CodaBench[2] platform, which facilitated access to the dataset, submission management, leaderboard evaluation, and standardized comparison of participating systems.

2.1 Dataset

For this **InLSum** Shared Task, we introduce the **InLSum** (**In**dian **L**egal **Sum**marization) dataset, which consists of court judgments from prominent Indian courts paired with expert-written abstractive summaries, both in English. To facilitate model development and evaluation, the dataset is divided into 3 splits: a *train* set of 1,200 datapoints, a *validation* set of 200 datapoints, and a *test* set of 400 datapoints. Each datapoint refers to a (judgment–summary) pair. Many judgments span several thousand words, whereas the summaries are substantially shorter, requiring high compression. Detailed statistics, including average document length, average summary length, and compression ratios (the ratio between the length of the summaries to that of the corresponding judgments), are reported in Table 1.

2.2 Evaluation

System outputs in the L-SUMM Shared Task were evaluated using these 3 widely adopted automatic relevance metrics: ROUGE-2, ROUGE-L, and BLEU.

ROUGE (Lin, 2004): It stands for *Recall-Oriented Understudy for Gisting Evaluation*, a crucial metric for assessing the n-gram overlap between the model-generated summaries and the reference summaries. In this task, In this study, we employed the ROUGE-2 score for measuring the bi-gram textual overlap and ROUGE-L for measuring the longest matching sequence of words using the Longest Common Subsequence (LCS) between the model-generated and reference summaries.

BLEU (Papineni et al., 2002): It stands for *Bilingual Evaluation Understudy*. It measures overlap between model-generated and reference summaries by considering n-gram-based precision.

To ensure uniform comparability across metrics, all scores were scaled to the range $[0, 100]$ (with higher values indicating better performance). The final leaderboard ranking was determined by computing the arithmetic mean of the F1-scores of ROUGE-2, ROUGE-L, and BLEU, for each system.

3 Shared Task Submissions

The L-SUMM Shared Task attracted a total of 9 participating teams. Brief summaries of the modeling approaches taken by these teams are described in this section.

[2] https://www.codabench.org/

Juris-Summ (Sheik et al., 2025). This system adopts an adaptive pipeline built upon the *Longformer Encoder–Decoder* (*LED*)[3] model (Beltagy et al., 2020). For shorter judgments (<8,000 words), LED is applied in a single-pass setting, while longer documents are processed hierarchically: they are split into overlapping 5,000-word chunks (20% overlap) that are individually summarized, after which a meta-summarization stage fuses the chunk summaries into a coherent final output. This strategy leverages LED's long-context capabilities while mitigating boundary effects and preserving global coherence.

BLANCKED (Parada et al., 2025). This team adopts a hybrid extractive-abstractive pipeline designed for extremely long legal judgments. Their approach first segments documents into 512-token semantic chunks and applies the extractive summarization method, *PACSUM* (Zheng and Lapata, 2019), to rank and select the most salient content, yielding a 1000-token condensed extract. This extract is then passed to the proprietary LLM, *Gemini-2.5-Pro*, for zero-shot abstractive refinement, utilizing optimized prompts to enhance coherence and minimize redundancy. The method balances efficiency with coverage, with the LLM primarily enhancing fluency rather than adding new content, and achieves consistent gains over a purely extractive baseline.

TLDR-Uniandes (Chica, 2025). This team investigates a spectrum of prompting strategies with multi-agent architectures over widely popular proprietary LLM, *GPT-4.1*. They evaluate multiple prompt families, progressing from simple `TL;DR` baselines to structured few-shot instructions that enforce target length, retention of legal terms, and n-gram density. Their strongest results come from a Reward System prompt, which incorporates progressive rewards for long exact spans, bonuses for legal transition phrases (e.g., 'held that', 'dismissed the appeal'), contextual multipliers for sentence placement optimizing for ROUGE and BLEU simultaneously. Beyond prompting, they design three multi-agent pipelines: a two-stage extract–abstract workflow using verbatim extraction; a domain-aware pipeline where legal-domain classification informs the structure and content emphasis of later stages; and an expansive 20-stage sequential pipeline that processes ten legal rhetorical roles (facts, arguments, reasoning, orders, citations, etc.) with dedicated extraction and abstraction agents. A synthesis agent then fuses all partial outputs into a coherent final summary, prioritizing high-fidelity n-gram matching. This combination of reward-driven prompting and modular multi-agent flows represents one of the interesting approaches in the shared task.

BDS-Lab (Sonowal and Sadhu, 2025). This team introduces a *Structure-Aware Chunking* (SAC) pipeline that explicitly aligns the summarization process with the rhetorical structure of legal judgments. Their method segments each document into *Facts*, *Arguments & Analysis*, and *Conclusion* using either (i) a heuristic rule-based system (SAC-H) built from lexical indicators observed in the training corpus or (ii) a zero-shot LLM-based segmentation method (SAC-LLM) using *Gemini-2.5-Pro*, which identifies section boundaries through structured prompts. After segmentation, the system allocates the summarization token limits proportionally across sections, leveraging empirical distributions of section lengths in gold summaries. Each segment is then summarized independently using *Legal-Pegasus*[4] and concatenated to form the final output. Their analysis highlights a key trade-off: while section-aligned chunking improves global coherence (ROUGE-L), it can reduce local n-gram fluency (ROUGE-2), revealing structural constraints inherent in long-document summarization.

SCaLAR (D and Madasamy, 2025). This team explored three systems built on the *Legal-Pegasus* model, addressing extreme document length through hierarchical summarization. Their baseline system employs naive token-based chunking (1000 tokens with overlap) followed by recursive summarization to aggregate local summaries into a final global one. The next system improves this process through *rhetorical chunking*: a BERT-based classifier assigns one rhetorical role to each sentence, and sentences sharing the same role are grouped into coherent semantic units, which are then summarized hierarchically using a role-aware fine-tuned version of the same model. System-3 extends from System-2 by incorporating *weighted rhetorical roles*, where each role is assigned an explicit importance score derived from its prominence in reference summaries of the *train* split; these scores are inserted into the input tags during fine-tuning. Collectively, their experiments investi-

[3] https://huggingface.co/allenai/led-large-16384

[4] https://huggingface.co/nsi319/legal-pegasus

Rank	Team Name	ROUGE-2↑	ROUGE-L↑	BLEU↑	AVG↑
1	FourCorners	34.91	33.34	21.49	**29.91**
2	Juris-Summ	29.62	28.56	21.67	**26.62**
3	TLDR-Uniandes	26.88	27.38	19.49	**24.58**
4	Contextors	25.13	25.59	16.80	**22.51**
5	GenAI-Lab	25.90	24.95	13.05	**21.30**
6	SCaLAR	21.86	25.93	14.43	**20.74**
7	BLANCKED	21.05	24.35	15.12	**20.17**
8	LegalAI	20.37	22.49	13.67	**18.84**
9	BDS-Lab	16.51	22.41	05.08	**14.67**

Table 2: Final Leaderboard results of the L-SUMM Shared Task in JUST-NLP 2025. AVG denotes the mean of ROUGE-2, ROUGE-L, and BLEU scores, where higher values (↑) indicate better performance.

gate how rhetorical structure and role-level importance signals influence the quality of summaries for long legal judgments.

LegalAI (Sha et al., 2025). This team explored two approaches: (i) a hybrid extractive–abstractive pipeline combining a *BART*-based model (fine-tuned on CNN/DailyMail and further adapted on InLSum) with *TextRank*-based extractive pre-selection, and (ii) a purely abstractive summarization approach using *Indian_Legal_Pegasus*[5], a domain-adapted variant of *Legal-Pegasus* fine-tuned on the Indian legal domain. In the first approach, sentence embeddings from *all-MiniLM-L6-v2* are used to build a similarity graph, and the top 20 TextRank-selected sentences are fed to BART (within a 1024-token limit). The second approach directly fine-tunes and applies the model under similar input and output length constraints. They illustrate the differences between extractive-guided and fully abstractive strategies for long-form legal summarization.

FourCorners (Chaksangchaichot and Akarajaradwong, 2025). This team presents a multi-stage alignment pipeline tuned by Reinforcement Learning (RL) built on *Qwen3-4B-Instruct-2507*[6] that ultimately secured the *top position* on the L-SUMM Shared Task leaderboard. Their pipeline begins with data filtering based on regression between judgment and summary lengths to remove noisy pairs. The model, *Qwen3-4B-Instruct-2507*, is then adapted using a two-stage supervised finetuning (SFT) strategy: an initial high-rank LoRA phase on medium-length inputs to shape task-specific behavior, followed by a long-context finetuning phase (10k–30k tokens) with lightweight adapters. A final Reinforcement Learning with Verifiable Rewards (RLVR) step further aligns generation quality by directly optimizing BLEU and ROUGE using **G**roup **S**equence **P**olicy **O**ptimization (GSPO) (Zheng et al., 2025) or **D**ecoupled Clip and **D**ynamic s**A**mpling **P**olicy **O**ptimization (DAPO) (Yu et al., 2025) style updates, sequence-level importance sampling, and low-rank LoRA for stable optimization. Finally, they demonstrate an RL-based approach to long-form legal summarization and show that their entire pipeline remains highly compute-efficient.

Contextors (Neelamegam and Nirmala, 2025). They fine-tune multiple pretrained Seq2Seq models (*BART*, *Legal-Pegasus*, *T5*, and *LED*) over InLSumm and integrate them through an ensemble framework. Pairwise and three-way ensembles (BART-Pegasus-LED) generate candidate summaries, which are then semantically ranked using *InLegalBERT* model to select the most contextually aligned output. They further propose a Retriever-Driven framework that identifies the most semantically relevant document chunks before fine-tuning each model and again uses semantic similarity for final summary selection.

GenAI-Lab (Jadav et al., 2025). This team also investigates retrieval–driven summarization for long legal documents, comparing three retrieval strategies – *Dense Passage Retrieval* (DPR) (Karpukhin et al., 2020), *Maximum Cosine Similarity* (MCS) (Shukla et al., 2022), and *Maximum Marginal Relevance* (MMR) (Xie and Liu, 2008) – to construct semantically aligned training pairs. Judgment texts are chunked into

[5] https://huggingface.co/akhilm97/pegasus_indian_legal
[6] https://huggingface.co/Qwen/Qwen3-4B-Instruct-2507

passages (e.g., 1024 tokens depending on the input context size of the model), and relevant passages are selected for each summary sentence using embedding-based similarity. Summaries are generated using both zero-shot LLMs (e.g., *Qwen-32B*, *GPT-OSS-120B*, *LlaMa-3*, and *LlaMa-4*) and fine-tuned encoder–decoder models, particularly LED and Legal-Pegasus. The fine-tuned models are trained on datasets produced by each retrieval method, enabling controlled evaluation of how retrieval quality impacts abstractive summarization performance.

4 Results and Findings

Table 2 presents the final leaderboard of the `L-SUMM` Shared Task. The results reveal clear performance variation across methodological families, reflecting the diversity of modeling strategies explored by participating teams.

The top-performing teams employ different strategies, including reinforcement learning (FourCorners), hierarchical long-context modeling (Juris-Summ), multi-agent prompting (TLDR-Uniandes), and retrieval-based multi-generator ensembles (Contextors and GenAI-Lab), while also emphasizing mechanisms that preserve global coherence in extremely long legal documents. The top performer, **FourCorners** (AVG: 29.91), couples supervised finetuning with GSPO/DAPO-based RL alignment, achieving the highest scores across major metrics despite using a relatively small (4B) model. Their results illustrate the effectiveness of reward optimization for long-form legal summarization. **Juris-Summ** (AVG: 26.62) and **TLDR-Uniandes** (AVG: 24.58) follow hierarchical LED summarization and multi-agent reward-driven prompting using GPT-4.1, respectively. The mid-ranking systems, including **Contextors** (AVG: 22.51) and **GenAI-Lab** (AVG: 21.30), employ retrieval-enhanced finetuning. Their moderate performance suggests that retrieval enhances factual grounding, although gains depend strongly on the choice of generator and selection heuristics. The lower-ranking systems, including **SCaLAR** (AVG: 20.74), **BLANCKED** (AVG: 20.17), **LegalAI** (AVG: 18.84), and **BDS-Lab** (AVG: 14.67), generally relied on more traditional pipelines such as extractive pre-processing or direct finetuning of legal-domain models.

Across systems, a central finding emerges: *handling document length and structure is more critical than the choice of backbone model*. Teams that explicitly address long-context challenges via hierarchical routing, retrieval augmentation, structured chunking, or reinforcement alignment, consistently outperform those relying solely on standard summarization architectures or zero-shot prompting.

5 Conclusion

The JUST-NLP 2025 `L-SUMM` Shared Task provides a benchmark for evaluating long-document summarization within the challenging domain of legal judgments. The task attracted 9 participating teams, who collectively explored a wide range of modeling paradigms, including long-context encoder-decoder architectures, retrieval-augmented pipelines, multi-agent prompting strategies, hybrid extractive-abstractive designs, and reinforcement learning–based alignment methods. The final leaderboard results demonstrate that the top-performing system leveraged a multi-stage SFT+RL pipeline to secure the leading position. Several other approaches have shown that retrieval quality, rhetorical role-based chunking strategies, and domain adaptation all play vital roles in shaping summary quality. Overall, the `L-SUMM` Shared Task highlights both the progress and the challenges that remain in long-form legal summarization. The benchmark, dataset resource, and system descriptions released through this shared task aim to support future research on long-form summarization in high-stakes domains, such as law. As long-context architectures, RL-strategies, alignment methods, and structure-aware techniques continue to improve, future iterations of this task are likely to further advance the capabilities of automated legal reasoning and summarization.

Limitations

In this shared task, our evaluation framework relies primarily on automatic metrics such as ROUGE and BLEU, which measure surface-level lexical overlap and provide only a syntactic assessment of summary quality. Semantic evaluation metrics (e.g., BERTScore) were not included in the official scoring pipeline. Also, the model-generated abstractive summaries are highly susceptible to factual inconsistencies, particularly when dealing with complex judicial reasoning and domain-specific terminology. However, our evaluation setup does not independently assess factual accuracy using factual consistency measures (e.g., SummaCONV or other hallucination-sensitive consistency metrics). Further, a major limitation is the absence of expert-driven human evaluation. Legal summarization requires domain knowledge, and expert review by legal professionals is essential for assessing correctness, completeness, and potential misinterpretations in generated summaries. Due to the high cost and limited availability of legal experts, we were unable to conduct a human evaluation phase. Consequently, the final leaderboard does not reflect expert assessments.

References

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *ArXiv*, abs/2004.05150.

Paheli Bhattacharya, Kaustubh Hiware, Subham Rajgaria, Nilay Pochhi, Kripabandhu Ghosh, and Saptarshi Ghosh. 2019. A comparative study of summarization algorithms applied to legal case judgments. In *Advances in Information Retrieval*, pages 413–428, Cham. Springer International Publishing.

Chompakorn Chaksangchaichot and Pawitsapak Akarajaradwong. 2025. A Budget Recipe for Finetuning a Long-form Legal Summarization Model. In *Proceedings of the First Workshop on NLP for Empowering Justice (JUST-NLP)*, Mumbai, India. Association for Computational Linguistics.

Santiago Chica. 2025. Automatic Legal Judgment Summarization Using Large Language Models: A Case Study for the JUST-NLP 2025 Shared Task. In *Proceedings of the First Workshop on NLP for Empowering Justice (JUST-NLP)*, Mumbai, India. Association for Computational Linguistics.

Arjun T D and Anand Kumar Madasamy. 2025. SCaLAR_NITK @ JUSTNLP Legal Summarization (L-SUMM) Shared Task. In *Proceedings of the First Workshop on NLP for Empowering Justice (JUST-NLP)*, Mumbai, India. Association for Computational Linguistics.

Debtanu Datta, Shubham Soni, Rajdeep Mukherjee, and Saptarshi Ghosh. 2023. MILDSum: A novel benchmark dataset for multilingual summarization of Indian legal case judgments. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5291–5302, Singapore. Association for Computational Linguistics.

Aniket Deroy, Kripabandhu Ghosh, and Saptarshi Ghosh. 2025. Applicability of large language models and generative models for legal case judgement summarization. *Artificial Intelligence and Law*, 33(4):1007–1050.

Nita Jadav, Ashok Urlana, and Pruthwik Mishra. 2025. NIT-Surat@L-Sum: A Semantic Retrieval-Based Framework for Summarizing Indian Judicial Documents. In *Proceedings of the First Workshop on NLP for Empowering Justice (JUST-NLP)*, Mumbai, India. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Pavithra Neelamegam and S Jaya Nirmala. 2025. Contextors at L-SUMM: Retriever-Driven Multi-Generator Summarization. In *Proceedings of the First Workshop on NLP for Empowering Justice (JUST-NLP)*, Mumbai, India. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Erich Giusseppe Soto Parada, Carlos Manuel Muñoz Almeida, and David Cuevas Alba. 2025. Combining Extractive and Generative Methods for Legal Summarization: BLANCKED at JUST-NLP 2025. In *Proceedings of the First Workshop on NLP for Empowering Justice (JUST-NLP)*, Mumbai, India. Association for Computational Linguistics.

Sayed Ayaan Ahmed Sha, Sangeetha Sivanesan, Anand Kumar Madasamy, and Navya Binu. 2025. Integrating Graph based Algorithm and Transformer Models for Abstractive Summarization. In *Proceedings of the First Workshop on NLP for Empowering Justice (JUST-NLP)*, Mumbai, India. Association for Computational Linguistics.

Saloni Sharma, Surabhi Srivastava, Pradeepika Verma, Anshul Verma, and Sachchida Nand Chaurasia. 2023. A comprehensive analysis of indian legal documents summarization techniques. *SN Comput. Sci.*, 4(5).

Reshma Sheik, Noah John Puthayathu, Fathima Firose A, and Jonathan Paul. 2025. Hierarchical Long-Document Summarization using LED for Legal Judgments. In *Proceedings of the First Workshop on NLP for Empowering Justice (JUST-NLP)*, Mumbai, India. Association for Computational Linguistics.

Abhay Shukla, Paheli Bhattacharya, Soham Poddar, Rajdeep Mukherjee, Kripabandhu Ghosh, Pawan Goyal, and Saptarshi Ghosh. 2022. Legal case document summarization: Extractive and abstractive methods and their evaluation. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1048–1064, Online only. Association for Computational Linguistics.

Himadri Sonowal and Saisab Sadhu. 2025. Structure-Aware Chunking for Abstractive Summarization of Long Legal Documents. In *Proceedings of the First Workshop on NLP for Empowering Justice (JUST-NLP)*, Mumbai, India. Association for Computational Linguistics.

Shasha Xie and Yang Liu. 2008. Using corpus and knowledge-based similarity measure in maximum marginal relevance for meeting summarization. In *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4985–4988.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, and 16 others. 2025. Dapo: An open-source llm reinforcement learning system at scale. *ArXiv*, abs/2503.14476.

Chujie Zheng, Shixuan Liu, Mingze Li, Xionghui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. 2025. Group sequence policy optimization. *ArXiv*, abs/2507.18071.

Hao Zheng and Mirella Lapata. 2019. Sentence centrality revisited for unsupervised summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6236–6247, Florence, Italy. Association for Computational Linguistics.

Linwu Zhong, Ziyi Zhong, Zinian Zhao, Siyuan Wang, Kevin D. Ashley, and Matthias Grabmair. 2019. Automatic summarization of legal decisions using iterative masking of predictive sentences. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Law*, ICAIL '19, page 163–172, New York, NY, USA. Association for Computing Machinery.

Findings of the JUST-NLP 2025 Shared Task on English-to-Hindi Legal Machine Translation

Kshetrimayum Boynao Singh[1], Sandeep Kumar[1], Debtanu Datta[2], Abhinav Joshi[3], Shivani Mishra[3], Shounak Paul[2], Pawan Goyal[2], Sarika Jain[4], Saptarshi Ghosh[2], Ashutosh Modi[3], Asif Ekbal[1]

[1]IIT Patna, India [2]IIT Kharagpur, India [3]IIT Kanpur, India [4]NIT Kurukshetra, India

Correspondence: boynfrancis@gmail.com

Abstract

This paper provides an overview of the Shared Task on Legal Machine Translation (L-MT), organized as part of the JUST-NLP 2025 Workshop at IJCNLP-AACL 2025, aimed at improving the translation of legal texts, a domain where precision, structural faithfulness, and terminology preservation are essential. The training set comprises 50,000 sentences, with 5,000 sentences each for the validation and test sets. The submissions employed strategies such as: domain-adaptive fine-tuning of multilingual models, QLoRA-based parameter-efficient adaptation, curriculum-guided supervised training, reinforcement learning with verifiable MT metrics, and from-scratch Transformer training. The systems are evaluated based on BLEU, METEOR, TER, chrF++, BERTScore, and COMET metrics. We also combine the scores of these metrics to give an average score (AutoRank). The top-performing system is based on a fine-tuned distilled NLLB-200 model and achieved the highest AutoRank score of 72.1. Domain adaptation consistently yielded substantial improvements over baseline models, and precision-focused rewards proved especially effective for the legal MT. The findings also highlight that large multilingual Transformers can deliver accurate and reliable English-to-Hindi legal translations when carefully fine-tuned on legal data, advancing the broader goal of improving access to justice in multilingual settings.

1 Introduction

The JustNLP 2025[1] Shared Task on English-to-Hindi Legal Machine Translation (L-MT)[2] was launched to address a critical need in India's multilingual legal system, where more than 44 million cases remain pending, and a significant portion of the judicial process still operates in English. This linguistic mismatch restricts accessibility for Hindi-speaking citizens, highlighting the need for reliable, domain-specific translation tools. Legal texts, however, are complex, characterized by dense syntax, specialized terminology, and strict semantic precision, making legal MT (Joshi et al., 2024) substantially more challenging than general-domain translation.

To foster progress in this area, the shared task provided a curated parallel corpus of Indian legal documents and a standardized evaluation framework. The participating teams explored a wide range of approaches, including multilingual Transformer fine-tuning, QLoRA-based parameter-efficient methods (Jeon and Strube, 2025), curriculum learning, reinforcement learning with verifiable rewards, and training models from scratch (Singh et al., 2023). This overview summarizes their methodologies and results, offering insight into current capabilities and emerging strategies for high-quality legal translation (Mahapatra et al., 2025). The broader goal of the initiative is to stimulate innovation in Indian-language NLP and identify promising, yet underexplored techniques (Dabre and Kunchukuttan, 2024) that can support scalable, accessible, and accurate legal document processing for Indian languages.

Our primary objective in organizing this collaborative effort is to bring together researchers and developers to explore effective strategies for enhancing the quality of Indian-language machine translation, particularly in the legal domain. A secondary objective was to identify interesting but still underexplored practices, even if they do not directly contribute to achieving state-of-the-art MT performance.

[1]https://exploration-lab.github.io/JUST-NLP/
[2]https://www.codabench.org/competitions/10351/

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 12–17
December 24, 2025

2 Task Dataset

2.1 Dataset for Fine-Tuning and Evaluation

For the development and assessment of the model, we introduce our English-to-Hindi legal parallel corpus, which serves as a training and validation dataset. This dataset consists of 50,000 sentence pairs for training[3], along with 5,000 validation sentence pairs. For testing, we evaluate on the WMT25 Legal benchmark dataset (Singh et al., 2025b), which includes 5,000 additional test sentences. To ensure that the model is trained on accurate and high-quality parallel data, a series of pre-processing and filtration procedures was implemented.

Our training corpus is curated specifically for the legal domain and encompasses a diverse range of judicial document types, including court judgments, legal petitions, bail applications, government regulations, and case proceedings. The average sentence length in the training data is approximately 29 words for English and 31 words for Hindi.

Filtering by Sentence Length. We excluded sentence pairs in which either the source or target sentence contained fewer than 5 words or more than 70 words in the training and validation sets. Sentences that are excessively short often provide minimal supervision signals, whereas overly long sentences introduce structural complexity that may compromise training stability. Therefore, this length-based filtering helps preserve a more uniform and trainable distribution of training samples.

Dataset Split	Language	Sentences	Words
Train	English	50,000	1,463,911
	Hindi	50,000	1,577,884
Validation	English	5,000	145,063
	Hindi	5,000	156,850
Test	English	5,000	134,016
	Hindi	5,000	133,471

Table 1: Statistics of the English and Hindi Legal Translation Dataset

Word Count Ratio Filtering. To further enhance the alignment quality, we filtered sentence pairings based on the ratio of word counts between the source and target sentences. Only pairings in which the difference in word count fell within the range of $[-10, +10]$ were preserved. This phase helps remove misaligned pairs where the target sentence is significantly longer or shorter than the source, thereby reducing noise and promoting improved structural correspondence between the languages.

[3] https://huggingface.co/datasets/helloboyn/IJCNLP-JustNLP-LMT

3 Evaluation Metrics

We use six widely used automatic evaluation metrics to fully assess system performance in the English-to-Hindi Legal MT shared task. Each metric examines a different aspect of translation quality, so that systems are evaluated not only on their surface similarity, but also on their accuracy in terms of meaning and context, which are crucial for legal translation.

- **BLEU (Papineni et al., 2002)**: Measures n-gram overlap between the system output and reference translation, focusing on lexical accuracy.
- **METEOR (Banerjee and Lavie, 2005)**: Evaluates translations using exact, stem, and synonym matches, offering a more flexible lexical comparison than BLEU.
- **TER (Snover et al., 2006)**: Computes the number of edits required to transform the MT output into the reference; lower scores indicate better structural alignment.
- **chrF++ (Popović, 2017)**: A character and word-level F-score metric that captures fine-grained morphological and orthographic similarities.
- **BERTScore (Zhang et al., 2020)**: Uses contextual embeddings to measure semantic similarity between the reference and the generated translation.
- **COMET (Rei et al., 2020)**: A neural metric trained on human judgments that captures adequacy and fluency using encoder–decoder representations.

Together, these metrics provide complementary perspectives, lexical, structural, semantic, and contextual, offering a more complete understanding of translation quality.

3.1 AutoRank for Unified Scoring

Each metric provides useful insights, but relying on only one can overemphasize certain translation characteristics, such as n-gram precision or semantic similarity. To avoid this kind of bias, we utilize AutoRank (Kocmi et al., 2025), a single scoring system that combines all six evaluation metrics into a single balanced score.

$$\text{AutoRank} = \frac{1}{6}\sum_{i=1}^{6} M_{i,\text{norm}} \quad (1)$$

AutoRank score[4] ensures that no single metric is more important than the others by placing all of them on the same scale. Metrics like BLEU, METEOR, chrF++, BERTScore, and COMET are scaled directly, whereas TER (where lower is better) is inverted to maintain consistency. The final AutoRank value is then found by averaging the normalized values on a scale ranging from 0 to 100.

This unified scoring method ensures that systems are evaluated as a whole, taking into account lexical accuracy, semantic meaning, structural fidelity, and contextual alignment, rather than relying on a single metric. Because of this, AutoRank provides the legal MT shared task with a fair, comprehensive, and dependable ranking system.

4 Summary of Participant Systems

A total of 24 teams initially registered for the shared task. Of these, 7 teams submitted system outputs, and 5 teams provided accompanying system description papers. The participating teams and their submitted systems are summarized as follows.

4.1 Team-SVNIT: Domain-Adaptive Fine-Tuning of Multilingual Models for English-Hindi Legal Machine Translation

Team-SVNIT (Dhakad et al., 2025) achieved 1st place in the shared task. Legal translation between English and Hindi is particularly challenging due to domain-specific terminology and long, syntactically complex sentence structures. To address this, the team fine-tunes and evaluates several multilingual pre-trained translation models, including `facebook/nllb-200-distilled-1.3B`, on the 50,000 English-Hindi legal sentence pairs provided by the organizers. Their training pipeline incorporates careful pre-processing, a 512-token context window, and optimized decoding strategies to improve robustness and translation fidelity. The final system attained an AutoRank score of 72.10, securing the top position on the leaderboard. Across evaluation metrics, the model achieved: BLEU 51.61, METEOR 75.80, TER 37.09, CHRF++ 73.29, BERTScore 92.61, and COMET 76.36. These results underscore the effectiveness of domain-adaptive fine-tuning for specialized legal MT tasks. The authors have publicly released their implementation for further research and reproducibility[5].

4.2 FourCorners: Cold Starts and Hard Cases - A Two-Stage SFT–RLVR Approach for Legal Machine Translation

FourCorners (Akarajaradwong and Chaksangchaichot, 2025) introduced one of the most innovative methodologies in the shared task. Their system for the JUST-NLP 2025 English–Hindi Legal Machine Translation challenge is built around a two-stage, data-centric training pipeline designed to handle both easy and difficult examples effectively. In the first stage, the team annotates the training corpus by translation difficulty, creating subsets labeled as ‘easy’, ‘medium’, and ‘hard’. Supervised fine-tuning (SFT) is then applied to the easy-to-medium partition to establish a strong cold start model. In the second stage, the authors employ Reinforcement Learning with Verifiable Rewards (RLVR) exclusively on the hard subset. The reward signal, derived from standard machine translation metrics such as BLEU, ROUGE-L, and chrF++, enables the model to optimize directly for precision-oriented translation quality. This two-stage SFT-RLVR framework yields substantial improvements over strong SFT baselines, validating the effectiveness of combining difficulty-aware data curation with metric-driven reinforcement learning for high-fidelity legal translation. The system ranked 2nd overall in the official leaderboard. The authors have released their code and model weights[6].

4.3 goodmen: A Comparative Study of Neural Models for English-Hindi Legal Machine Translation

The *goodmen* (K et al., 2025) team ranked 3rd overall in the shared task. In a multilingual country like India, ensuring that legal judgments are accessible in native languages is essential for equitable

[4] https://huggingface.co/datasets/helloboyn/IJCNLP-JustNLP-LMT/blob/main/JustNLP25_L-MT_Result_Final.pdf

[5] https://github.com/Rupeshddhakad06/JUST-NLP-LMT

[6] https://github.com/ppaolong/FourCorners-JustNLP-MT-Shared-Task

justice. The Legal Machine Translation (L-MT) shared task aims to address this need by focusing on English-Hindi legal translation. The authors present a systematic comparative evaluation of neural MT models for this domain, examining four multilingual and Indic-focused systems developed under the shared-task constraints.

Their approach emphasizes domain-specific fine-tuning while preserving statutory structure, legal citations, and jurisdictional terminology. In particular, they fine-tune two legal-oriented models: `InLegalTrans` and `IndicTrans2` on the 50k English-Hindi legal parallel corpus provided by the organizers, with no external data permitted. The fine-tuned `InLegalTrans` model achieves the strongest performance, reaching a BLEU score of 48.56, substantially outperforming its base version. The comparative study reveals that targeted domain adaptation substantially enhances translation quality for specialized legal texts. Human evaluation further confirms that the fine-tuned `InLegalTrans` outputs better preserve legal coherence and judicial tone. The team's best-performing model is ranked 3rd on the test set. The authors have released their model[7].

4.4 JUNLP: From Scratch to Fine-Tuned: A Comparative Study of Transformer Training Strategies for Legal Machine Translation

The JUNLP (Barman et al., 2025) team ranked 4th overall in the shared task. In multilingual countries like India, access to legal information is often constrained by linguistic barriers, as much of the judicial and administrative discourse remains in English. Legal Machine Translation (L-MT) offers a scalable approach to accessible legal communication by enabling the consistent and accurate translation of legal texts. This work, developed for the JUST-NLP 2025 shared task, investigates English–Hindi legal translation using Transformer-based approaches. The authors evaluate two complementary strategies: fine-tuning a pre-trained OPUS-MT model for domain adaptation, and training a Transformer model from scratch using only the provided legal corpus. These systems are assessed across a broad suite of MT metrics, including SacreBLEU, chrF++, TER, ROUGE, BERTScore, METEOR, and COMET. Their fine-tuned OPUS-MT model achieves a SacreBLEU score of 46.03, substantially outperforming both the baseline and from-scratch models. The results demonstrate the clear advantage of domain adaptation for legal MT, showing that pretrained models fine-tuned on in-domain data significantly surpass models trained from scratch. The study highlights the potential of L–MT systems to improve legal accessibility and transparency in multilingual settings[8].

4.5 JUST-MEI: Adapting IndicTrans2 for Legal Domain MT via QLoRA Fine-Tuning

The JUST-MEI (Singh et al., 2025a) system ranked 5th overall in the shared task. Legal Machine Translation presents unique difficulties due to domain-specific terminology, long and formally structured statutes, and the high precision required for legal communication. As part of the JUST-NLP 2025 Shared Task on English-Hindi legal translation, the authors adapt the pre-trained `ai4bharat/indictrans2-en-indic-1B` model using QLoRA, a parameter-efficient fine-tuning strategy designed for low-resource computational settings. Using only the domain-specific parallel corpus provided by the organizers, the fine-tuned model achieves substantial improvements over the baseline IndicTrans2 system, especially in handling specialized legal vocabulary and complex syntactic constructions. In automatic evaluation, the system obtains a BLEU score of 46.67 and a chrF++ score of 70.03. Human evaluation further reflects strong performance, with adequacy and fluency scores of 4.085 and 4.006, respectively. The approach achieves a final AutoRank score of 67.98, demonstrating the effectiveness of QLoRA-driven domain adaptation for legal MT[9].

5 Results and Findings

Among the five participating systems, JUST-NLP 2025 Shared Task on English-to-Hindi Legal Machine Translation revealed that domain-adaptive fine-tuning remains the most decisive factor for achieving high-quality legal translations. The top-ranked 'Team-SVNIT' demonstrated that large multilingual models, such as `facebook/nllb-200-distilled-1.3B`, when carefully fine-tuned with long-context training, rigorous preprocessing, and stable optimization, de-

[7] https://huggingface.co/drjk16/InLegalTrans-Finetuned-JUSTNLP2025

[8] https://github.com/atanumandal0491/Legal-Translation

[9] https://drive.google.com/drive/folders/1USkOkqvV3HxFILAPsFpkxWYJ1FjNFRUl

Rank	Team Name	BLEU↑	METEOR↑	TER↓	chrF++↑	BERTScore↑	COMET↑	AutoRank↑
1	Team-SVNIT	51.61	75.80	37.09	73.29	92.61	76.36	72.10
2	FourCorners	50.19	69.54	42.32	73.67	92.70	75.74	69.92
3	goodmen	48.56	67.15	41.63	73.07	92.38	75.16	69.12
4	JUNLP	46.03	71.84	42.08	70.59	91.19	73.72	68.55
5	JUST-MEI	46.67	72.86	44.63	70.03	90.86	72.12	67.98
6	Lawgorithms	46.27	71.80	43.06	68.32	91.03	72.14	67.75
7	Tokenizers	34.08	61.78	55.25	56.75	87.39	65.20	58.32

Table 2: Evaluation results for the English-to-Hindi Legal MT Shared Task.

liver state-of-the-art performance across lexical and semantic metrics. 'FourCorners' contributed the most innovative approach, showing that curriculum-based learning combined with reinforcement learning using verifiable MT metrics can outperform conventional supervised fine-tuning, especially in precision-sensitive domains like law. The 'goodmen' team highlighted the value of domain-specific architectures, with their fine-tuned `InLegalTrans` model excelling in preserving statutory structure and legal terminology. 'JUNLP' results further underscored the superiority of pretrained models, as their fine-tuned OPUS-MT system significantly outperformed a Transformer trained from scratch. Finally, 'JUST-MEI' demonstrated that parameter-efficient adaptation via QLoRA can achieve competitive performance even under limited computational resources. Together, these findings affirm that legal MT requires not only linguistic fluency but rigorous domain adaptation, precision-oriented optimization, and careful engineering, marking a substantial advancement in the development of reliable English-to-Hindi legal translation systems.

6 Conclusion

The shared task demonstrates that high-quality English-to-Hindi legal translation is well within reach when modern Transformer models are carefully tailored to the legal domain. We found that simply using large, pre-trained models is not enough; fine-tuning them on legal data is essential for capturing the precision and structure that legal texts require. Approaches that prioritize accuracy over paraphrasing, especially those optimized with BLEU and chrF++, consistently produced more reliable outputs. Innovative strategies, such as combining supervised training with reinforcement learning, further boosted performance, and even lightweight methods like QLoRA proved effective for teams working with limited compute resources. Overall, these results demonstrate that with the right combination of domain adaptation, careful engineering, and precision-focused training, today's MT systems can produce translations that are accurate, trustworthy, and suitable for real-world legal applications.

Limitations

The shared task is limited only to the English-to-Hindi direction and does not consider the reverse Hindi-to-English direction. Although we follow the AutoRank framework with normalized scores for fair system comparison, the inclusion of TER as an inverted (negative) metric may not always correlate consistently with other positive evaluation metrics, which could potentially influence the final ranking. Finally, the shared task evaluation is conducted only at the sentence level; however, a document-level evaluation would be essential for capturing broader contextual dependencies and discourse-level translation quality, particularly in the legal domain.

Acknowledgement

The authors express gratitude to the COIL-D Project under Bhashini, funded by MeitY, Govt. of India, for the support and resources that facilitated the successful execution of this research.

References

Pawitsapak Akarajaradwong and Chompakorn Chaksangchaichot. 2025. Cold Starts and Hard Cases: A Two-Stage SFT-RLVR Approach for Legal Machine Translation (Just-NLP L-MT shared task). In *Proceedings of the First Workshop on NLP for Empowering Justice (JUST-NLP)*, Mumbai, India. Association for Computational Linguistics.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Pro-*

ceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Amit Barman, Atanu Mandal, and Sudip Kumar Naskar. 2025. From Scratch to Fine-Tuned: A Comparative Study of Transformer Training Strategies for Legal Machine Translation. In *Proceedings of the First Workshop on NLP for Empowering Justice (JUST-NLP)*, Mumbai, India. Association for Computational Linguistics.

Raj Dabre and Anoop Kunchukuttan. 2024. Findings of WMT 2024's MultiIndic22MT shared task for machine translation of 22 Indian languages. In *Proceedings of the Ninth Conference on Machine Translation*, pages 669–676, Miami, Florida, USA. Association for Computational Linguistics.

Rupesh Dhakad, Naveen Kumar, and Shrikant Malviya. 2025. Team-SVNIT at JUST-NLP 2025: Domain-Adaptive Fine-Tuning of Multilingual Models for English–Hindi Legal Machine Translation. In *Proceedings of the First Workshop on NLP for Empowering Justice (JUST-NLP)*, Mumbai, India. Association for Computational Linguistics.

Sungho Jeon and Michael Strube. 2025. Entity tracking in small language models: An attention-based study of parameter-efficient fine-tuning. In *Proceedings of the 6th Workshop on Computational Approaches to Discourse, Context and Document-Level Inferences (CODI 2025)*, pages 42–53, Suzhou, China. Association for Computational Linguistics.

Abhinav Joshi, Shounak Paul, Akshat Sharma, Pawan Goyal, Saptarshi Ghosh, and Ashutosh Modi. 2024. IL-TUR: Benchmark for Indian legal text understanding and reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11460–11499, Bangkok, Thailand. Association for Computational Linguistics.

Deeraj S K, Karthik Suryanarayanan, Yash Ingle, and Pruthwik Mishra. 2025. goodmen @ L-MT Shared Task: A Comparative Study of Neural Models for English-Hindi Legal Machine Translation. In *Proceedings of the First Workshop on NLP for Empowering Justice (JUST-NLP)*, Mumbai, India. Association for Computational Linguistics.

Tom Kocmi, Ekaterina Artemova, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Konstantin Dranch, Anton Dvorkovich, Sergey Dukanov, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Marzena Karpinska, Philipp Koehn, Howard Lakougna, Jessica Lundin, Christof Monz, Kenton Murray, and 10 others. 2025. Findings of the WMT25 general machine translation shared task: Time to stop evaluating on easy test sets. In *Proceedings of the Tenth Conference on Machine Translation*, pages 355–413, Suzhou, China. Association for Computational Linguistics.

Sayan Mahapatra, Debtanu Datta, Shubham Soni, Adrijit Goswami, and Saptarshi Ghosh. 2025. Milpac: A novel benchmark for evaluating translation of legal text to indian languages. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 24(8).

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Maja Popović. 2017. chrF++: words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation*, pages 612–618, Copenhagen, Denmark. Association for Computational Linguistics.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.

Akoijam Jenil Singh, Loitongbam Sanayai Meetei, and Yumnam Surajkanta. 2025a. Adapting IndicTrans2 for Legal Domain MT via QLoRA Fine-Tuning at JUST-NLP 2025. In *Proceedings of the First Workshop on NLP for Empowering Justice (JUST-NLP)*, Mumbai, India. Association for Computational Linguistics.

Kshetrimayum Boynao Singh, Deepak Kumar, and Asif Ekbal. 2025b. Evaluation of LLM for English to Hindi legal domain machine translation systems. In *Proceedings of the Tenth Conference on Machine Translation*, pages 823–833, Suzhou, China. Association for Computational Linguistics.

Kshetrimayum Boynao Singh, Ningthoujam Avichandra Singh, Loitongbam Sanayai Meetei, Sivaji Bandyopadhyay, and Thoudam Doren Singh. 2023. NITS-CNLP low-resource neural machine translation systems of English-Manipuri language pair. In *Proceedings of the Eighth Conference on Machine Translation*, pages 967–971, Singapore. Association for Computational Linguistics.

Matthew Snover, Bonnie Dorr, Rich Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 223–231, Cambridge, Massachusetts, USA. Association for Machine Translation in the Americas.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. *Preprint*, arXiv:1904.09675.

LeCNet: A Legal Citation Network Benchmark Dataset

Pooja Harde
National Institute of Technology Kurukshetra
Kurukshetra, Haryana, India
pmharde29@gmail.com

Bhavya Jain
Indian Institute of Technology Bhilai
Durg, Chhattisgarh, India
bhavyaj@iitbhilai.ac.in

Sarika Jain
National Institute of Technology Kurukshetra
Kurukshetra, Haryana, India
jasarika@nitkkr.ac.in

Abstract

Legal document analysis is pivotal in modern judicial systems, particularly for case retrieval, classification, and recommendation tasks. Graph neural networks (GNNs) have revolutionized legal use cases by enabling the efficient analysis of complex relationships. Although existing legal citation network datasets have significantly advanced research in this domain, the lack of large-scale open-source datasets tailored to the Indian judicial system has limited progress. To address this gap, we present the Indian Legal Citation Network (LeCNet) - the first open-source benchmark dataset for the link prediction task (missing citation recommendation) in the Indian judicial context. The dataset has been created by extracting information from the original judgments. LeCNet comprises 26,308 nodes representing case judgments and 67,108 edges representing citation relationships between the case nodes. Each node is described with rich features of document embeddings that incorporate contextual information from the case documents. Baseline experiments using various machine learning models were conducted for dataset validation. The Mean Reciprocal Rank (MRR) metric is used for model evaluation. The results obtained demonstrate the utility of the LeCNet dataset, highlighting the advantages of graph-based representations over purely textual models.

1 Introduction

The rapid advancement of artificial intelligence (AI) has revolutionized numerous fields, including law, where the automation of legal document analysis has emerged as a critical area of research. Efficient retrieval, classification, and recommendation of legal cases are essential for assisting lawyers, judges, and legal researchers in navigating vast repositories of case law. However, developing AI-based systems for legal analysis demands high-quality datasets tailored to specific jurisdictions, along with robust methodologies to address unique challenges in legal text understanding and reasoning.

The increasing volume of cases adjudicated by judiciaries poses significant challenges in maintaining decision-making accuracy, consistency, and fairness (Varghese, 2024). Legal professionals, including lawyers and judges, require efficient tools to reduce the time and cost associated with legal research, improving the effectiveness and quality of the judicial process. Legal professionals traditionally rely on their expertise and critical thinking to identify and refer relevant prior cases when addressing a specific legal case (query case). While technology has been introduced to assist in this process, its role has largely been limited to rudimentary tools such as keyword searches and Boolean operations. Additionally, with the rapid growth of legal case databases, even the most experienced practitioners find it progressively more difficult to efficiently locate and cite pertinent past cases (noticed cases) (Joshi et al., 2023).

Incorporating machine learning into the recommendation of legal citations is a key step in utilizing AI to assist legal experts. Citations are essential in legal writing, particularly within common law systems, where they support arguments by referencing statutes, administrative regulations, and case law that interprets these laws in various scenarios. The importance of citations is particularly evident in legal education, where the process of selecting law journal editors often includes a stringent evaluation of citation formatting skills (Cross and Spriggs, 2010). Additionally, excelling in more complex tasks like legal text generation and summarization requires a deep understanding of citations, highlighting their crucial role in legal discourse.

(Huang et al., 2021) explores the application of link prediction with the context of legal citation recommendation, also sometimes referred to as

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 18–28

prior case retrieval (Kano et al., 2019; Joshi et al., 2023), to identify and suggest relevant legal precedents or references using citation networks in legal documents. In general, there are many open domain datasets available for the link prediction task, such as the citation network of articles (Wang et al., 2024), arxiv papers by MAG (Wang et al., 2024), papers100M (Wang et al., 2024), wikikg2 (Vrandečić and Krötzsch, 2014), collab (Wang et al., 2024), etc. Datasets that can be used for the legal citation recommendation task are available for US[1], German(Milz et al., 2021), and Canadian cases (COLIEE)(Kano et al., 2019), IL-PCR (Joshi et al., 2023). Similar works have also been carried out for the Indian judiciary by Paheli et. al. (Bhattacharya et al., 2022) and Khatri et. al. (Khatri et al., 2023). The main limitations of existing work are (1) the absence of a legal dataset specifically designed for the link prediction task and (2) the unavailability of large-scale legal citation datasets tailored to the Indian judiciary. The Supreme Court of India's hierarchical, multi-tier system (including High Courts) yields a sparse, hierarchical network where a few landmark cases dominate citations. These patterns differ from US or German legal networks and from prior document-similarity datasets. The existing datasets raise two research questions and leave them unaddressed. **RQ1:** How do dataset design choices (e.g. number of citations removed or negative sampling strategy) affect model performance and robustness? **RQ2:** What is the impact of using contextual document embeddings (versus other text features) on citation link-prediction in graph based models?

We aim to address the stated research questions and introduce a large-scale Indian Legal Citation Network dataset called LeCNet. The proposed dataset is the first of its kind developed for the Indian judiciary, where legal judgments are represented as nodes and citation relationships between cases as edges ($case \xrightarrow{cites} case$), creating a rich graphical representation of 26,308 nodes and 67,108 directed edges. Beyond structural representation in the form of nodes and edges, the dataset also incorporates advanced node features in the form of document embeddings generated by the Doc2Vec model. LeCNet is designed to benchmark standard citation link-prediction models, including both non-graph and graph-based models. This work advances legal research by offering the following contributions:

- Considering the lack of available benchmarks for the Indian legal setting, we create a new benchmark of Legal Citation Network for the Indian legal system (LeCNet) (section 3).
- We have performed experiments on different LeCNet configurations to determine the best-performing version for the link prediction task.
- We have performed the model-based validation of LeCNet using non-GraphML and GraphML models to understand the structural and contextual dependency for the legal citation recommendation task (section 5.3).

2 Related Work

Researchers have explored various approaches to the challenges of legal domain analytics, such as (Kalamkar et al., 2022; Brugman, 2018; Filtz, 2017; Tang et al., 2020), to improve the efficiency and precision of legal workflows. The field of legal document analysis has witnessed significant progress in recent years, driven by advances in natural language processing (NLP) and graph-based machine learning.

Legal citation recommendation has emerged as a key task in the streamlining of legal research, intending to help legal professionals identify relevant precedents and legal references. However, network-based methods primarily focus on the structural information of the constructed network and tend to overlook the textual content of legal documents. Additionally, these methods are less effective when the network is sparse (Liu and Hsu, 2019). Various commercial tools have been developed to aid legal research through citation-based functionalities. Zhang and Koppaka (Zhang and Koppaka, 2007) highlight a LexisNexis feature that facilitates navigation within a semantic citation network by utilizing textual similarity between citation contexts. Similarly, platforms such as Thomson Reuters' CoCounsel (formerly known as CaseText's CARA A.I.) (CoCounsel), Parallel Search (2020), and Thomson Reuters' Quick Check (Thomas et al., 2020) offer citation recommendation services, though the underlying techniques remain undisclosed. Winkels et al. (Winkels et al., 2014) propose a recommender system tailored to Dutch immigration law, which retrieves cases with

[1] https://www.courtlistener.com/

high between-ness centrality relative to selected legal provisions. Dadgostari et al. (Dadgostari et al., 2021) address the challenge of creating bibliographies for citation-free legal texts by modeling the process as a Markov Decision Problem, where an agent iteratively identifies relevant documents using Q-learning, outperforming simpler approaches for retrieving U.S. Supreme Court decisions. Other researchers (Fowler et al., 2007; Koniaris et al., 2017) have studied the structure of legal citation networks, exploring metrics such as authority, relevance, and network properties like degree distribution and shortest path lengths. Sadeghian et al. (Sadeghian et al., 2018) introduce a system to extract and classify legal citations, predicting their purpose based on predefined labels. Huang et al. (Huang et al., 2021) investigate the use of deep learning techniques like BiLSTM and RoBERTa for legal citation prediction, showing that integrating contextual information enhances accuracy and providing benchmarks for advancing legal natural language processing research.

Legal citation network serve as the foundational structure for citation recommendation systems. These networks represent legal documents (nodes) and their interconnections through citations (edges). In the scientific community, researchers have used citation networks for recommendations in many different domains, such as e-commerce, commercial, and academics (Wang et al., 2024).

In 2011, Kumar et al. (Kumar et al., 2011) inferred document similarity by calculating the Jaccard similarity index between sets of out-citations and in-citations within document clusters, known as Bibliographic Coupling and Cocitation. In 2015, Minocha et al. (Minocha et al., 2015) assessed whether sets of precedent citations (out-citations) appeared in the same cluster to determine document similarity. Liu (Liu, 2017) improved upon Bibliographic Coupling in 2017 by incorporating the titles of out-citation references, thus enriching the model's informational context. In 2020, Bhattacharya et al. (Bhattacharya et al., 2020) applied Node2Vec (Grover and Leskovec, 2016) to map legal cases into vector embeddings, evaluating similarity based on these embeddings. Further, in 2022, Bhattacharya et al. (Bhattacharya et al., 2022) acknowledged the critical role of legal statute hierarchy and integrated it into a heterogeneous graph alongside legal case documents. Pioneering contributions include those by James Fowler et. al. (Fowler et al., 2007) and subsequent efforts by researchers such as Katz et. al. (Katz et al., 2020) and Mike Bommarito et. al. (Bommarito II et al., 2010). Additionally, Hoadley et al. (Hoadley et al., 2021) conducted a large-scale international citation network analysis, although their commercial dataset is not publicly available. This broader ecosystem highlights that legal citation datasets have evolved considerably, offering both structural and textual modalities for research.

The relational structure of legal citation networks makes them well-suited for Graph Machine Learning (GraphML) techniques. In **Graph Machine Learning for Legal Citation Networks**, legal documents, statutes, and related entities are represented as nodes, while edges denote relationships such as cites, relatedTo, or refersTo. This graph-based representation enables the application of powerful models like Graph Neural Networks (GNNs), Graph Convolutional Networks (GCNs), and Graph Attention Networks (GATs), which have demonstrated effectiveness in key tasks such as link prediction, node classification, and representation learning. Recent efforts, such as the Open Graph Benchmark introduced by Hu et al. (Hu et al., 2020), have catalyzed progress in GraphML by providing standardized datasets and evaluation protocols.

Problem Statement

In the legal research community, the area of GraphML and related graph-based datasets is still underexplored. The above-mentioned works concentrate on the effectiveness of legal citation networks; however, the unavailability of datasets and the limited focus on applications in the GraphML domain remain unaddressed. The prior works highly consider the node classification task, which may not be suitable for learning the legal cases representation in the embedding space for a citation network. To spur research in legal analytics, we introduce a large-scale Indian Legal Case Citation Network (LeCNet) gold standard dataset that will serve as a benchmark for the link prediction task. This will further aid in downstream applications like legal citation recommendations, similar case recommendations, etc.

3 Legal Citation Network (LeCNet) Benchmark Dataset

The Legal Citation Network (LeCNet) Benchmark Dataset is a dataset of Supreme Court of India court case judgments in English containing 26308

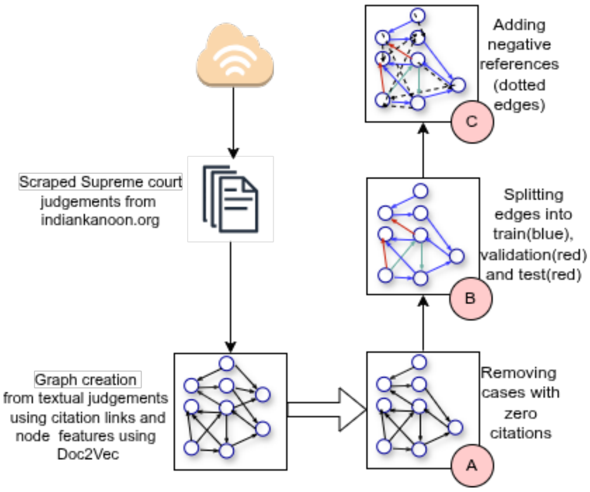


Figure 1: Designing of LeCNet dataset from scraping to graph creation to preprocessing to edge splitting and finally addition of negative references

legal documents. The case citing other cases are connected via an edge between the nodes, thus creating a directed graph. The mathematical annotation of the dataset can be given as:

LeCNet is a directed graph $G = (V, E)$ where V represents the set of nodes (SCI judgment documents) and $E \subseteq V \times V$ represents the set of edges (citations within the corpus). The LeCNet dataset construction is a three-step process, as described in the following subsections.

3.1 Corpus Creation and Preprocessing

The first step begins with the corpus creation. The corpus is created by scraping legal judgment documents (in the public domain) from the IndianKanoon[2] website. While the site provides access to legal information, it doesn't have a specific section detailing copyright terms for the entire website. The Supreme Court of India (SCI) judgments are simplified in nature as compared to the other Indian courts (High Courts and Apex Courts). A benchmark dataset starts with clean subsets before incorporating more complex structures. So, the SCI has been used for the initial release as a strength to provide a focused, consistent, and reproducibile benchmark before tackling the noisier High Court or tribunal data. In future releases, we plan to extend LeCNet with hierarchical (Supreme–High Court), cross-jurisdictional, and temporal data, making the retrieval challenge progressively more realistic and complex.

[2] https://indiankanoon.org/

We picked documents of the Supreme Court of India (SCI) cases ranging from the year 1950 to 2022 (considered as zero-hop set cases) using a custom Python-based web crawler library Selenium[3]. (Malik et al., 2021)). To gather citation cases, we scraped the documents appearing in the immediate citation neighborhood of each source paper. During the collection of citations, only SCI citations are considered. The created corpus is then preprocessed to provide a unique number to every document to represent as a node in the graph. The data was validated through structured checks guided by six law students and two faculty from Law Department of Kurukshetra University, Haryana. For example, we confirmed the court levels (focusing on Supreme Court cases), set the date ranges, and removed any cases with zero remaining citations (Refer Fig1 A).

Node Features. The node features in the LeCNet dataset are derived from 100-dimensional Doc2Vec embeddings trained on the textual descriptions or metadata associated with each node. This ensures semantically similar nodes have embeddings that are close in the feature space, allowing effective incorporation of textual information into the graph for tasks like link prediction and node classification. The 100-dimensional size balances representational capacity with computational efficiency.

Doc2Vec was chosen for its ability to capture semantic relationships across entire documents, making it well-suited for complex, domain-specific data like legal texts. Prior studies (Mandal et al., 2021; Zhang and Zhou, 2019; Bhattacharya et al., 2022) highlight its superiority over Word2Vec, LegalBERT and Transformer-based models in representing the nuanced structure of legal documents. They confirmed that Doc2Vec outperformed other models for this domain, validating its use for generating node features in the LeCNet dataset.

3.2 Edge Selection for Dataset Splitting

Further, the dataset has been split into the train/test/validation sets for model development. We assume that all the available tuples form the training set in the beginning. The task is to select the edges to be moved to the test and the validation sets. To handle cases, where nodes have a small number of outgoing citations (e.g., one or two citations), we employed a data-splitting strategy such that if x outgoing citations are randomly

[3] https://selenium-python.readthedocs.io/

moved from the training set for creating a test and a validation set, then at least one outgoing edge will be retained in the training set for that source node (Refer Fig1 B). Additionally, the x selected edges are equally divided between the test and the validation sets; therefore, $x = 2n, \quad n \in \mathbb{N}$. An optimal value of x has to be decided and is done during parameter setting (5.2). The steps for the data splitting are described in Algorithm 1.

Algorithm 1 Edge Selection for Data Splitting

1: Take some $x = 2n, \quad n \in \mathbb{N}$. We use x=2 and x=4 to ensure validation and test have enough samples.

2: We select source cases based on the number of outgoing citations they contain:

$$V_{\text{source}} = \{s \in V \mid |E(s)| > x\},$$

where V is the set of all cases, and $E(s)$ is the set of edges emanating from source case s.

3: Repeat for all $s \in V_{\text{source}}$:

(i) From $E(s)$, randomly select x edges and mark them:

$$\{e_1^+, \ldots, e_x^+\} \subseteq E(s).$$

(ii) Move marked edges to validation/testing:

$$E(s)_{\text{valid}} \subseteq \{e_1^+, \ldots, e_x^+\}$$
$$|E(s)_{\text{valid}}| = \lfloor \tfrac{x}{2} \rfloor$$
$$E(s)_{\text{test}} = \{e_1^+, \ldots, e_x^+\} \setminus E(s)_{\text{valid}}$$

(iii) Remaining edges go to training:

$$E(s)_{\text{train}} = E(s) \setminus \{e_1^+, \ldots, e_x^+\}.$$

3.3 Selection of Target Negative References:

The purpose of negative references is to provide a contrastive learning signal for the model. Here, the model learns why certain references exist and why others do not. Without negative examples, the model might simply learn to predict any link between cases, regardless of their content or context. Hence, using negative references, the model learns to identify the features that distinguish between true (positive) citations and carefully selected negative citations. Negative samples for the nodes are generated by first marking a constrained set of potential negative references and then randomly selecting NF references for every source node as depicted in Algorithm 2 and Fig1 C. An optimal value of NF has to be decided and is done during parameter setting 5.2.

Algorithm 2 Negative Reference Sampling

Repeat for all $s \in V_{\text{source}}$:

(i) Define the candidate set $V^-(s)$, consisting of all nodes not directly cited by s:

$$V^-(s) = \{v \in V \mid (s, v) \notin E\},$$

where V is the set of all cases and E is the set of citation edges.

(ii) From $V^-(s)$, randomly select NF nodes to form the negative reference set for s.:

$$E^-(s) \subseteq V^-(s), \qquad |E^-(s)| = \text{NF}.$$

Thus, the model is trained to predict the missing references and rank them above the negative candidates, ensuring an effective citation recommendation system.

4 Link Prediction Task

Our Link Prediction Task is essentially a Missing Citation Recommendation Task where for each source node, we drop some citations and aim to rank these missing citations (i.e., outgoing edges). To unlock new frontiers for legal researchers in the area of Graph Machine Learning (GraphML) for legal citation recommendation, we consider three non-GraphML and five GraphML models as our baselines for validating the LeCNet dataset. Both the choice of baselines and the ranking formulation (positives ranked above negatives) follow the standard link-prediction setup introduced by (Hu et al., 2020). Legal citations often exhibit relevant legal context characteristics that these models are well-equipped to capture. While this work primarily serves as a resource-oriented benchmark, the chosen models are theoretically grounded and practically relevant for modeling citation behavior in legal texts. Below, we describe how each model obtains node embeddings:

Non-GraphML Models:

- **MLP:** Input node features are directly used as node embeddings.

- **NODE2VEC (N2V):** The node embeddings are obtained by concatenating input features and NODE2VEC embeddings (Grover and Leskovec, 2016; Perozzi et al., 2014).
- **MATRIX FACTORIZATION (MAT. FACT.:** The distinct embeddings are assigned to different nodes and are learned in an end-to-end manner together with the MLP predictor.

GraphML Models:

- **GCN:** The node embeddings are obtained by full-batch Graph Convolutional Networks (GCN) (Kipf and Welling, 2016).
- **GRAPHSAGE (G.SAGE):** The node embeddings are obtained by full-batch GraphSAGE (Hamilton et al., 2017), where we adopt its mean pooling variant and a simple skip connection to preserve central node features.
- **NEIGHBOR SAMPLING (N. SAMP.):** A mini-batch training technique of GNNs that samples neighborhood nodes when performing aggregation (Hamilton et al., 2017).
- **CLUSTERGCN (C.GCN):** A mini-batch training technique of GNNs that partitions the graphs into a fixed number of subgraphs and draws mini-batches from them (Chiang et al., 2019).
- **GRAPHSAINT (G.SAINT):** A mini-batch training technique of GNNs that samples subgraphs via a random walk sampler (Zeng et al., 2019).

4.1 Evaluation Metric

We evaluated model performance using Mean Reciprocal Rank (MRR), a standard metric for ranking tasks such as citation prediction using the link prediction task. For each source case s, the reciprocal rank of a true reference e_i^+ among all candidate cases is defined as:

$$\text{Reciprocal Rank}(s, e_i^+) = \frac{1}{\text{rank of } e_i^+ \text{ among } e^-(s)} \quad (1)$$

The MRR is computed by averaging the reciprocal ranks over all source cases S:

$$\text{MRR} = \frac{1}{|S|} \sum_{s \in S} \frac{1}{\text{rank of } e^+ \text{ among } e^-(s)} \quad (2)$$

MRR captures the position of the first correct citation in the ranked list, aligning well with practical legal retrieval scenarios where users typically seek the most relevant precedent quickly. Its sensitivity to top-ranked predictions makes it particularly suitable for evaluating the utility of citation recommendation models in a legal context. We report MRR scores to ensure comparability with prior work (Hu et al., 2020) and to reflect the effectiveness of our models in producing useful citation suggestions.

5 Dataset Validation

Once the dataset was developed as in section 3, we performed its performance tuning and model-based validation.

5.1 Computational Setup and Reproducibity details

Each experiment was conducted across 10 runs and the best result out of them was considered. The models were implemented in PyTorch v2.5.1 and run using CUDA v12.4 on an NVIDIA GeForce RTX 3060 GPU. We tuned hyperparameters manually based on performance on the validation set given in Table1. All models evaluated in our experiments were trained using Adam optimizer and had 3 layers in their architecture. Model selection was based on the best performance on the validation set. No hyperparameters were tuned using the test set. Specific packages used for graph data processing and model implementation include torch-geometric for GCN, GraphSAGE, GraphSAINT, and Cluster-GCN and node2vec from the stellargraph/torch_geometric.nn.models library. We release the dataset in the best-performing dataset configuration for research use via Drive. The license for the uploaded content is Creative Commons Attribution 4.0 International (CC-BY). We intend that this dataset be used for research purposes in various tasks, such as legal citation recommendation and prior case retrieval. Our implementation of the baseline models follows the general training and evaluation outlined in (Hu et al., 2020) so we do not release separate model-training code and refer readers to the original implementations for reproducibility.

5.2 LeCNet Dataset Performance Tuning

To rigorously analyze the performance and robustness of citation recommendation models on the LeCNet dataset, we conducted studies focusing on two key tunable parameters: (1) the number of outgoing citation edges (denoted as x) to be moved to the test and validation sets, and (2) the number of target negative reference candidates per source

Model	**Learning Rate**	**Epochs**	**Additional Hyperparameters**
MLP	0.01	100	–
N2V	0.01	100	Emb. Dim.=128, Walk len.=40, Walks per node=10
MAT.FAC.	0.01	150	Emb. Dim.=96
G.SAGE	0.001	50	–
N.SAMP.	0.005	150	–
C.GCN	0.001	100	–
G.SAINT	0.001	100	Walk length = 3, Num steps = 100
GCN	0.001	50	–

Table 1: Best-found hyperparameters for different models.

node (denoted as NF). These ablations enabled us to systematically evaluate model behavior under varying levels of structural sparsity and negative sampling complexity. For detailed mean and standard deviation values corresponding to each dataset configuration, one can refer to the tables provided in the Annexure.

5.2.1 Determining Optimal x

The first experiment is based on the edge selection for the dataset splitting into train, test, and validation sets as mentioned in 3.2. The number of negative reference candidates is kept fixed at ($DS_{NF=10}$), while the number of removed outgoing edges is varied ($DS_{x=2}$ and $DS_{x=4}$) to evaluate performance under increasing information sparsity. Algorithm 1 describes how the two dataset configurations $DS_{x\,=\,2}$ and $DS_{x\,=\,4}$ were created for the experiment whose statistics are described in Table 2. These strategies ensure that even for low-degree nodes, sufficient information is preserved in the training set for meaningful learning while still providing test and validation examples for evaluation.

Table 3 *(the first two out of the three major columns)* present the results of running various Machine Learning models on **LeCNet** for the link prediction task for the two dataset configurations to be tested. Contrary to initial expectations, increasing the number of removed citations (from 2 to 4) does not significantly degrade performance for most models as models have access to more structural information. It suggests that withholding a larger number of edges limits the structural information available during training. At the same time, the performance gap between the two settings is relatively small for most models, indicating that increasing x does not severely degrade performance. This balance implies that while the models remain robust even when more edges are shifted out of the training graph, $x = 2$ provides a more favourable trade-off between training signal and evaluation size, and is therefore the more suitable choice for subsequent experiments.

Dataset Config.	E_{train}	E_{valid}	E_{test}
$DS_{\text{x=2}}$	51186	7961	7961
$DS_{\text{x=4}}$	50760	8174	8174

Table 2: Statistics for no. of edges in train, validation, and test sets for the two dataset configurations, one with x=2, the other with x=4

5.2.2 Determining Optimal NF

Another experiment is based on the impact of adding a different number of target negative references to the citation network. We add NF target negative references for each source node in the validation and test sets and the aim of the model is to rank the true missing x citations higher than NF negative references. The negative references are randomly-sampled from all the previous cases that are not cited by the source node.

In this experiment, the number of removed edges is kept fixed ($DS_{x=2}$) *(as decided in section 5.1.1)*, and the number of negative references is varied ($DS_{NF=10}$ and $DS_{NF=25}$) to analyze how models perform under increasing difficulty in identifying correct links from negative references.

Table 3 *(the last two out of the three major columns)* present the results of running various Machine Learning models on **LeCNet** for the link prediction task for the two dataset configurations to be tested. Increasing the number of negative references (from 10 to 25) reduces performance reflecting the increased difficulty of distinguishing true citations from a larger pool of distractors. Notably, GCN maintains performance more effectively than the other architectures, indicating that it retains discriminative capability even when the neg-

Models	$DS_{\text{x=4, NF=10}}$			$DS_{\text{x=2, NF=10}}$			$DS_{\text{x=2,NF=25}}$		
	Train	**Validation**	**Test**	**Train**	**Validation**	**Test**	**Train**	**Validation**	**Test**
MLP	0.838	0.807	0.807	0.830	0.832	0.831	0.734	0.742	0.737
N2V	0.887	0.856	0.857	0.886	0.874	0.873	0.800	0.796	0.791
MAT. FACT.	0.998	0.373	0.372	0.998	0.378	0.377	0.99	0.382	0.380
G.SAGE	0.992	0.979	0.980	0.991	0.985	0.984	0.980	0.968	0.968
N. SAMP.	0.982	0.981	0.982	0.980	0.987	0.985	0.959	0.970	0.970
C.GCN	0.919	0.907	0.909	0.916	0.929	0.926	0.861	0.877	0.875
G.SAINT	0.959	0.955	0.953	0.953	0.963	0.963	0.916	0.933	0.933
GCN	0.991	**0.983**	**0.983**	0.989	**0.988**	**0.987**	0.989	**0.988**	**0.987**

Table 3: MRR performance of Open Graph Benchmark models on LeCNet across three dataset configurations: $DS_{x=4,NF=10}$, $DS_{x=2,NF=10}$, and $DS_{x=2,NF=25}$

ative set becomes substantially larger. $NF = 10$ provides a balanced and reliable configuration for evaluating link prediction performance.

5.3 Model-based Validation

The evaluation on several link prediction models helps us understand the dataset's suitability for link prediction tasks and its comparative performance against established benchmarks. The evaluation of various models on the LeCNet dataset under different configurations highlights the effectiveness of graph-based approaches for legal citation recommendation. Refer to the column 2 *($DS_{x=2,NF=10}$)* of Table 3 for the model-based validation of the LeCNet, as this is the most optimal dataset configuration.

We observe that GCN consistently outperforms other models, achieving the highest Mean Reciprocal Rank (MRR) scores in both validation and testing phases. The performance of NeighborSampling and GraphSAGE is also notable, demonstrating their ability to capture structural dependencies effectively. In contrast, Matrix Factorization exhibits severe overfitting, performing exceptionally well during training but failing to generalize, as indicated by its significantly lower validation and testing scores. Traditional approaches like MLP and Node2Vec lag behind graph-based models, reaffirming the importance of leveraging graph neural networks (GNNs) for this task.

Across varying graph configurations too, GCN demonstrates superior scalability and generalization, making it the most effective model for legal citation prediction.

6 Conclusion

Our findings supplement the significance of graph-based models in learning from citation networks, demonstrating their superiority over traditional methods. Specifically, GCN and GraphSAGE emerge as the most effective models on LeCNet, achieving high Mean Reciprocal Rank (MRR) scores across different data splits. Their ability to capture structural dependencies in citation networks allows for more accurate link prediction compared to non-graph-based approaches like MLP and Node2Vec. Additionally, this study underscores the importance of evaluating models under varying levels of network sparsity, as real-world citation graphs often suffer from missing or incomplete links. Ensuring robustness under such conditions will be crucial for deploying citation prediction models in practical legal applications.

Despite these strong performances, there remains room for further enhancements. We plan to incorporate the paragraph-level citation recommendation (e.g., case facts, citation reasoning, etc.) into the dataset as future work.

Limitations

In this paper, we evaluated the LeCNet dataset using Open Benchmark Dataset models for citation prediction. While the results demonstrate the effectiveness of GNN-based methods and scalable mini-batch training techniques, there remains significant room for improvement. One limitation of our approach is the reliance on official citations as the ground truth, which may not always capture the full scope of relevant citations due to the subjective nature of citation practices in legal writing. Exploring alternative ground truth definitions that account for implicit relevance could improve model performance.

Additionally, our evaluation primarily focused on MRR as the metric, which, while effective, might not fully capture other aspects of citation prediction, such as diversity or contextual relevance. Incorporating alternative metrics could provide a more nuanced understanding of model performance.

Another limitation lies in the dataset itself. The dataset contains only the citations from the Supreme Court of India cases, whereas a case might cite other judicial cases, also like High Courts, which are not considered during dataset creation. Incorporating citations across other courts or jurisdictions (e.g., High Court cases citing Supreme Court decisions) is an important extension that we plan for future releases. LeCNet is intended primarily as a benchmark for citation link-prediction methods in the Indian legal domain, and secondarily as training data for legal retrieval tasks. Note that our current evaluation assumes a fixed corpus; supporting truly new (unseen) documents is an important direction for future work. In future work, we plan to involve legal scholars for expert feedback to further validate the dataset's relevance and the models' real-world applicability.

An important direction for future work is to incorporate harder negative samples, such as cases from similar legal topics or issued in the same year. Such negatives would better reflect the real ambiguity present in citation decisions and create a more challenging and realistic evaluation setting for link prediction models.

Finally, while we conducted extensive experiments using existing Open Graph Benchmark models, further exploration of advanced techniques, such as contrastive learning, hybrid GNNs, or the inclusion of richer features like legal concepts, statutes, or temporal patterns, could enhance the task of citation prediction. Future work could also investigate models capable of incorporating domain-specific knowledge to address the unique challenges presented by citation prediction in the legal domain.

Ethical Considerations

This paper releases a dataset for recommending relevant citations in the legal domain. The purpose is not to replace researchers or legal experts, but to contribute a dataset that can be used to augment their work by facilitating efficient citation recommendations. For training and evaluation, we utilized the LeCNet dataset, which consists of publicly available citation data, ensuring that privacy concerns are not violated.

It is important to note that we did not explicitly address potential biases in the dataset or normalize the data concerning citation patterns. As such, the system's performance may reflect certain biases inherent in the data, such as overrepresentation of frequently cited papers or underrepresentation of niche topics. Future work could involve investigating these biases and exploring methods to ensure a more balanced and fair recommendation system.

References

CaseText 2020. Machine learning behind parallel search. Accessed: 2025-01-10.

Paheli Bhattacharya, Kripabandhu Ghosh, Arindam Pal, and Saptarshi Ghosh. 2020. Hier-spcnet: a legal statute hierarchy-based heterogeneous network for computing legal case document similarity. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*, pages 1657–1660.

Paheli Bhattacharya, Kripabandhu Ghosh, Arindam Pal, and Saptarshi Ghosh. 2022. Legal case document similarity: You need both network and text. *Information Processing & Management*, 59(6):103069.

Michael J Bommarito II, Daniel Martin Katz, Jonathan L Zelner, and James H Fowler. 2010. Distance measures for dynamic citation networks. *Physica A: Statistical Mechanics and its Applications*, 389(19):4201–4208.

Simon Brugman. 2018. Deep learning for legal tech: exploring ner on dutch court rulings. *Data Science Group Faculty of Science*.

Wei-Lin Chiang, Xuanqing Liu, Si Si, Yang Li, Samy Bengio, and Cho-Jui Hsieh. 2019. Cluster-gcn: An efficient algorithm for training deep and large graph convolutional networks. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 257–266.

CoCounsel. Cocounsel: One genai assistant for professionals. Accessed: 2025-01-10.

Frank B Cross and James F Spriggs. 2010. Citations in the us supreme court: An empirical study of their use and significance. *U. Ill. L. Rev.*, page 489.

Faraz Dadgostari, Mauricio Guim, Peter A Beling, Michael A Livermore, and Daniel N Rockmore. 2021. Modeling law search as prediction. *Artificial Intelligence and Law*, 29:3–34.

Erwin Filtz. 2017. Building and processing a knowledge-graph for legal data. In *The Semantic Web: 14th International Conference, ESWC 2017, Portorož, Slovenia, May 28–June 1, 2017, Proceedings, Part II 14*, pages 184–194. Springer.

James H Fowler, Timothy R Johnson, James F Spriggs II, Sangick Jeon, and Paul J Wahlbeck. 2007.

Network analysis and the law: Measuring the legal importance of precedents at the us supreme court. *Political Analysis*, 15(3):324–346.

Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864.

Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30.

Daniel Hoadley, M Bartolo, R Chesterman, A Faus, W Hernandez, B Kultys, AP Moore, E Nemsic, N Roche, J Shangguan, and 1 others. 2021. A global community of courts? modelling the use of persuasive authority as a complex network. *Frontiers in Physics*, 9:665719.

Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. 2020. Open graph benchmark: Datasets for machine learning on graphs. *Advances in neural information processing systems*, 33:22118–22133.

Zihan Huang, Charles Low, Mengqiu Teng, Hongyi Zhang, Daniel E Ho, Mark S Krass, and Matthias Grabmair. 2021. Context-aware legal citation recommendation using deep learning. In *Proceedings of the eighteenth international conference on artificial intelligence and law*, pages 79–88.

Abhinav Joshi, Akshat Sharma, Sai Kiran Tanikella, and Ashutosh Modi. 2023. U-creat: Unsupervised case retrieval using events extraction. *arXiv preprint arXiv:2307.05260*.

Prathamesh Kalamkar, Astha Agarwal, Aman Tiwari, Smita Gupta, Saurabh Karn, and Vivek Raghavan. 2022. Named entity recognition in Indian court judgments. In *Proceedings of the Natural Legal Language Processing Workshop 2022*, pages 184–193, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Yoshinobu Kano, Mi-Young Kim, Masaharu Yoshioka, Yao Lu, Juliano Rabelo, Naoki Kiyota, Randy Goebel, and Ken Satoh. 2019. Coliee-2018: Evaluation of the competition on legal information extraction and entailment. In *New Frontiers in Artificial Intelligence: JSAI-isAI 2018 Workshops, JURISIN, AI-Biz, SKL, LENLS, IDAA, Yokohama, Japan, November 12–14, 2018, Revised Selected Papers*, pages 177–192. Springer.

Daniel Martin Katz, Corinna Coupette, Janis Beckedorf, and Dirk Hartung. 2020. Complex societies and the growth of the law. *Scientific reports*, 10(1):18737.

Mann Khatri, Mirza Yusuf, Yaman Kumar, Rajiv Ratn Shah, and Ponnurangam Kumaraguru. 2023. Exploring graph neural networks for indian legal judgment prediction. *arXiv preprint arXiv:2310.12800*.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Marios Koniaris, Ioannis Anagnostopoulos, and Yannis Vassiliou. 2017. Network analysis in the legal domain: A complex model for european union legal sources. *Journal of Complex Networks*, 6(2):243–268.

Sushanta Kumar, P Krishna Reddy, V Balakista Reddy, and Aditya Singh. 2011. Similarity analysis of legal judgments. In *Proceedings of the fourth annual ACM Bangalore conference*, pages 1–4.

Rey-Long Liu. 2017. A new bibliographic coupling measure with descriptive capability. *Scientometrics*, 110:915–935.

Rey-Long Liu and Chih-Kai Hsu. 2019. Improving bibliographic coupling with category-based cocitation. *Applied Sciences*, 9(23):5176.

Vijit Malik, Rishabh Sanjay, Shubham Kumar Nigam, Kripa Ghosh, Shouvik Kumar Guha, Arnab Bhattacharya, and Ashutosh Modi. 2021. Ildc for cjpe: Indian legal documents corpus for court judgment prediction and explanation. *arXiv preprint arXiv:2105.13562*.

Arpan Mandal, Kripabandhu Ghosh, Saptarshi Ghosh, and Sekhar Mandal. 2021. Unsupervised approaches for measuring textual similarity between legal court case reports. *Artificial Intelligence and Law*, pages 1–35.

Tobias Milz, Michael Granitzer, and Jelena Mitrović. 2021. Analysis of a german legal citation network. In *International Conference on Knowledge Discovery and Information Retrieval*.

Akshay Minocha, Navjyoti Singh, and Arjit Srivastava. 2015. Finding relevant indian judgments using dispersion of citation network. In *Proceedings of the 24th international conference on World Wide Web*, pages 1085–1088.

Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710.

Ali Sadeghian, Laksshman Sundaram, Daisy Zhe Wang, William F. Hamilton, Karl Branting, and Craig Pfeifer. 2018. Automatic semantic edge labeling over legal citation graphs. *Artificial Intelligence and Law*, 26(2):127–144.

Mingwei Tang, Cui Su, Haihua Chen, Jingye Qu, and Junhua Ding. 2020. Salkg: a semantic annotation system for building a high-quality legal knowledge graph. In *2020 IEEE International Conference on Big Data (Big Data)*, pages 2153–2159. IEEE.

Merine Thomas, Thomas Vacek, Xin Shuai, Wenhui Liao, George Sanchez, Paras Sethia, Don Teo, Kanika Madan, and Tonya Custis. 2020. Quick check: A legal research recommendation system. In *Proceedings of NLLP '20*, volume 2645. CEUR-WS.

Dr John Varghese. 2024. Datafication in judicial case management in india. In *Symposium on Diversity in Legal and Judicial Profession and the Politics of Merit and Exclusion in India, RHUL, London.*

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Jie Wang, Kanha Bansal, Ioannis Arapakis, Xuri Ge, and Joemon M Jose. 2024. Empowering legal citation recommendation via efficient instruction-tuning of pre-trained language models. In *European Conference on Information Retrieval*, pages 310–324. Springer.

Radboud Winkels, Alexander Boer, Bart Vredebregt, and Alexander von Someren. 2014. Towards a legal recommender system. In *Proceedings of JURIX '14*.

Hanqing Zeng, Hongkuan Zhou, Ajitesh Srivastava, Rajgopal Kannan, and Viktor Prasanna. 2019. Graphsaint: Graph sampling based inductive learning method. *arXiv preprint arXiv:1907.04931*.

Haoyang Zhang and Liang Zhou. 2019. Similarity judgment of civil aviation regulations based on doc2vec deep learning algorithm. In *2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pages 1–8. IEEE.

Paul Zhang and Lavanya Koppaka. 2007. Semantics-based legal citation network. In *Proceedings of ICAIL '07*, pages 123–130.

A Appendix

A1. Detailed MRR Results for $DS_{x=4,NF=10}$

Models	Train	Validation	Test
MLP	0.838±0.005	0.807±0.002	0.807±0.001
N2V	0.887±0.005	0.856±0.001	0.857±0.001
MAT.FAC.	0.998±0.000	0.373±0.011	0.372±0.010
G.SAGE	0.992±0.000	0.979±0.000	0.980±0.001
N.SAMP.	0.982±0.000	0.981±0.000	0.982±0.000
C.GCN	0.919±0.002	0.907±0.003	0.909±0.003
G.SAINT	0.959±0.003	0.955±0.003	0.953±0.003
GCN	0.991±0.000	0.983±0.000	0.983±0.001

Table 4: MRR results for $DS_{x=4,NF=10}$.

A2. Detailed MRR Results for $DS_{x=2,NF=10}$

Models	Train	Validation	Test
MLP	0.830±0.004	0.832±0.001	0.831±0.002
N2V	0.886±0.007	0.874±0.001	0.873±0.001
MAT.FAC.	0.998±0.000	0.378±0.015	0.377±0.013
G.SAGE	0.991±0.001	0.985±0.000	0.984±0.001
N.SAMP.	0.980±0.001	0.987±0.000	0.985±0.000
C.GCN	0.916±0.005	0.929±0.003	0.926±0.004
G.SAINT	0.953±0.002	0.963±0.001	0.963±0.002
GCN	0.989±0.001	0.988±0.000	0.987±0.001

Table 5: MRR results for $DS_{x=2,NF=10}$.

A3. Detailed MRR Results for $DS_{x=2,NF=25}$

Models	Train	Validation	Test
MLP	0.734±0.007	0.742±0.002	0.737±0.001
N2V	0.800±0.004	0.796±0.002	0.791±0.002
MAT.FAC.	0.990±0.000	0.382±0.012	0.380±0.001
G.SAGE	0.980±0.001	0.968±0.000	0.968±0.001
N. SAMP.	0.959±0.001	0.970±0.001	0.970±0.001
C.GCN	0.861±0.007	0.877±0.007	0.875±0.007
G.SAINT	0.916±0.004	0.933±0.004	0.933±0.005
GCN	0.989±0.001	0.988±0.000	0.987±0.000

Table 6: MRR results for $DS_{x=2,NF=25}$.

Legal Document Summarization: A Zero-shot Modular Agentic Workflow Approach

Taha Sadikot
National Institute of Technology
Kurukshetra
taha.sadikot.m@gmail.com

Sarika Jain
National Institute of Technology
Kurukshetra
jasarika@nitkkr.ac.in

Abstract

The large volume and inherent complexity of Indian Court judgments, which feature nuanced legal arguments and extensive factual details, have created a need for high-quality automated summarization systems. We develop two zero-shot modular agentic workflow frameworks for Indian Court judgment summarization that do not require model fine-tuning: a three-stage Lexical Modular Summarizer (LexA) designed for lexical overlap metrics and a five-stage Semantic Agentic Summarizer (SemA) designed for semantic similarity. We extract a subset of CivilSum and IN-Abs datasets and call it the Sum-IPL-CivilSum test set. On this test set, LexA achieves ROUGE-1 F1 of 0.6326 and BERTScore F1 of 0.8902, comparable to state-of-the-art fine-tuned transformer models while requiring no training data or GPU resources. On the Sum-IPL-IN-Abs test set, LexA achieves ROUGE-1 F1 of 0.1951 and SemA achieves ROUGE-1 F1 of 0.2014 and BERTScore F1 above 0.81, outperforming zero-shot baselines. Our evaluation suggests that modular, zero-shot agentic approaches can achieve competitive results for legal summarization in resource-limited judicial settings.

1 Introduction

The Indian judicial system generates vast volumes of lengthy judgments, often exceeding 5,000 words, making manual summarization a significant bottleneck for legal professionals (Supreme Court of India, 2024; Malik et al., 2024). Current state-of-the-art solutions rely on fine-tuning transformer models like BART (Lewis et al., 2020) and Legal-BERT (Chalkidis et al., 2020), but these approaches are computationally expensive, require large annotated datasets, and lack the interpretability essential for legal trust. Additionally, their rigidity necessitates costly retraining when legal conventions evolve, limiting their utility for smaller stakeholders.

This work proposes zero-shot agentic workflows as a flexible alternative, guided by four core research questions. We investigate whether these workflows can match the performance of fine-tuned models (RQ1) and outperform direct LLM prompting (RQ2). Furthermore, we analyze whether architectural differences within the workflows produce meaningful performance trade-offs (RQ3) and assess their ability to generalize effectively across diverse legal datasets (RQ4).

Research Objective: To evaluate whether modular zero-shot agentic workflows can achieve competitive performance on automated summarization of Indian Court judgments without fine-tuning.

Key Contributions: We make the following key contributions in this paper:

1. **Zero-shot Modular Agentic Framework:** We develop a modular agentic framework for summarizing Indian Court judgments that operates in a zero-shot setting without supervised model fine-tuning. This work applies agentic workflow architectures to automated summarization of full-length judicial decisions.

2. **Competitive Empirical Performance:** We demonstrate empirically that this framework achieves ROUGE and BERTScore metrics at par with leading fine-tuned transformer baselines (ROUGE-1 F1: 0.6326 (Sum-IPL-CivilSum), BERTScore F1: 0.8902 (Sum-IPL-CivilSum)), achieving ROUGE-1 F1 of 0.6326 compared to 0.374 for Llama 2-chat-70B on the same benchmark. This modular architecture provides stepwise process transparency through explicit workflow decomposition and memory-based state management.

3. **Two Complementary Architectures:** We introduce and evaluate two distinct agentic workflow architectures: Lexical Modular Summarizer (LexA) (3-stage modular) and Semantic Agentic Summarizer (SemA) (5-stage

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 29–40
December 24, 2025 ©2025 Association for Computational Linguistics

integrated), demonstrating flexibility in design objective without model retraining.

4. **Comprehensive Evaluation and Analysis:** We evaluate our framework extensively using quantitative metrics, qualitative expert assessments, error analysis, and detailed workflow comparisons on our test set Sum-IPL, which is extracted from standard Indian legal datasets (CivilSum, IN-Abs).

Our proposed framework demonstrates that modular, agentic, and zero-shot approaches can be more practical and accessible for judicial systems facing resource constraints. The remainder of this paper is organized as follows: Section 2 provides background and related work. Section 3 details our methodology and the proposed workflow architectures. Section 4 describes the experimental setup. Section 5 presents comprehensive evaluation results, including quantitative performance, qualitative assessments, and error analysis. Section 6 discusses the implications, and Section 7 concludes with key findings and future directions.

2 Background and Related Work

Legal document summarization approaches have evolved from rule-based methods to modern agentic architectures. There have been recent advances in LLM-based reasoning and multi-agent systems that enable task decomposition. In this section, we review the relevant work in these dimensions.

2.1 Legal Document Summarization

Legal document summarization has evolved from early statistical methods to advanced neural architectures, with contemporary research focusing on fine-tuning transformers like BART and Legal-LED on specialized datasets such as IN-Abs and CivilSum. Despite achieving strong metrics, these models face adoption barriers due to high computational and data requirements. Consequently, the field is increasingly prioritizing broader challenges—including multi-granularity, legal reasoning, and multilingualism—highlighted by benchmarks like LegalBench and LEXTREME. Future directions emphasize developing robust methods capable of handling these complexities, particularly within the multilingual Indian legal context, without relying solely on resource-intensive training processes.

Our work explores whether agentic workflows can achieve comparable performance without these requirements.

2.2 Taxonomy of Text Summarization Approaches

Text summarization techniques can be broadly classified into various types (Jain and Saha, 2025; Smith and Wang, 2024).

2.2.1 Based on Summary Generation Strategy

Extractive Summarization involves selecting important sentences and phrases directly from the text and generating a concise summary (Brown and Taylor, 2022; Smith and Wang, 2024). This technique uses ranking algorithms to rank sentences, such as term frequency-inverse document frequency (TF-IDF), sentence position, and keyword occurrence. Classical examples of this approach include TextRank (Mihalcea and Tarau, 2004), LexRank (Erkan and Radev, 2004), and graph-based algorithms to preserve factual accuracy. However, this approach may result in potentially inconsistent output and may not convey the complete picture of the text.

Abstractive Summarization does not pick text from the main source, but generates new sentences that paraphrase and condense the main concepts of the source text, mimicking human summarization behavior (Gupta and Sharma, 2024). Modern abstractive approaches use encoder-decoder architectures, sequence-to-sequence models with attention mechanisms, and transformer-based models like BART (Lewis et al., 2020), PEGASUS (Zhang et al., 2020), and T5 (Raffel et al., 2020). Although abstractive summaries are more human-readable and consistent, they face challenges that include factual inconsistencies (hallucinations), difficulty maintaining legal precision, and higher computational requirements.

Hybrid Summarization combines the strengths of both extractive and abstractive methods (Patel and Singh, 2024). Typically, salient content from source documents is first extracted and then rewritten or paraphrased to improve consistency and readability (Patel and Singh, 2024). This approach balances the merits and demerits of both approaches.

2.2.2 Based on Implementation Methodology

Traditional Rule-based and Statistical Methods employ hand-crafted features, frequency-based heuristics, and graph algorithms (TextRank, LexRank) (Mihalcea and Tarau, 2004; Erkan and Radev, 2004) without machine learning. These approaches are highly interpretable but limited in their ability to handle complex linguistic patterns of legal texts.

Classical Machine Learning Methods involve early supervised models such as SVMs and decision trees. They use engineered features to predict which sentences to include in a summary.

Neural/Supervised Learning Methods use labeled datasets of document-summary pairs to train deep networks. They include classic sequence-to-sequence models and modern fine-tuned transformer models (e.g., BART (Lewis et al., 2020), T5 (Raffel et al., 2020), PEGASUS (Zhang et al., 2020), Legal-BERT (Chalkidis et al., 2020)). These also include pretrained and fine-tuned Large Language Models (LLMs), such as BERT and GPT. They achieve strong performance results both in extractive and abstractive summarization, but demand a significant amount of annotated data and computational resources for training (Jain and Saha, 2025; Smith and Wang, 2024).

Zero-shot, Few-shot, and Prompt-Based Methods operate without task-specific training data, relying on pre-trained general-purpose language models and prompt engineering. The emergence of instruction-tuned LLMs (GPT-3.5+, Claude, PaLM, Llama 2) (Chung et al., 2022) has enabled zero-shot application to specialized domains. In legal NLP, early results show promise: InstructGPT achieves 85% accuracy on CUAD clause extraction without fine-tuning (Hendrycks et al., 2021), and GPT-4 reaches 70% on LegalBench tasks (Guha et al., 2023). However, for legal summarization, Llama 2-chat-70B achieves only 0.374 ROUGE-1 on CivilSum (Malik et al., 2024), substantially below fine-tuned BART (0.450) on IN-Abs (Bhattacharya et al., 2019). These approaches offer flexibility and ease of deployment but have traditionally underperformed compared to supervised methods.

Agentic Workflow-based Methods represent a modular paradigm where summarization is divided into specialized subtasks, each managed by an agent (Chen and Zhang, 2024). Such systems are flexible and interpretable, combining elements like planning, tool selection, and iterative decision-making, all without requiring model retraining (Johnson and Lee, 2025; Chen and Zhang, 2024).

2.3 Agentic AI Systems: From Single Models to Multi-Agent Workflows

Recent advances in legal summarization have transitioned from monolithic models to multi-agent workflows that decompose complex tasks, leveraging foundational techniques like ReAct (Yao et al., 2023b) and Chain-of-Thought (CoT) (Wei et al., 2022) to improve performance by interleaving reasoning with actions. While methodologies like Tree of Thoughts (Yao et al., 2023a) or Reflexion (Shinn et al., 2023) explore exhaustive search or self-reflection, we adapt core reasoning strategies for targeted paragraph analysis and event extraction within a structured coordinator-executor architecture built on LangGraph (Team, 2024). Distinct from the complex peer-to-peer communication in CAMEL (Li et al., 2023) or dynamic routing in HuggingGPT (Shen et al., 2023), our approach prioritizes predictable production behavior by implementing specialized agent roles with persistent state management—drawing on the component taxonomy of Wang et al. (Wang et al., 2023) and standardized procedures similar to MetaGPT (Hong et al., 2024)—to ensure the verifiable outputs essential for legal applications.

Agentic AI systems can be classified into four levels based on their decision-making autonomy:

- **Level 1: Autonomous Agents** — Models that produce summaries from raw input entirely independently, with minimal external intervention. These agents are aspirational and currently limited to very controlled environments.

- **Level 2: Router/Coordination Workflows** — Modular systems where a core routing component assigns tasks (like fact extraction, event detection) to specialized agents that can act autonomously within predefined boundaries. This allows for powerful orchestration, easy insertion of new subtasks, and fine-grained error handling. Our workflows, Lexical Modular Summarizer (LexA) and Semantic Agentic Summarizer (SemA), operate at this level.

- **Level 3: Output Fusion Workflows** — Multiple summarization agents (e.g., extractive, abstractive, domain-specific models) generate intermediate outputs, which are then aggregated, ranked, or combined by a merging unit to maximize quality, diversity, or reliability.

- **Level 4: Human-in-the-Loop Workflows** — Systems that embed human expertise at key junctures, enabling legal or domain experts to review, correct, or validate intermediate or final outputs for more accountability, safety, and continuous improvement (Wilson and Kumar, 2024).

While agentic frameworks have been successfully applied to software development, their application to full-length legal document summarization remains unexplored. The legal domain presents unique challenges, including extreme length, complex argumentation, and strict accuracy requirements, making it an ideal testbed for evaluating whether modular, zero-shot workflows can match supervised approaches.

2.4 Positioning Our Approach

Our proposed framework operates as a Level 2 Router Workflow with hybrid extractive-abstractive summary generation characteristics. It is a zero-shot, prompt-based approach using general-purpose LLMs without fine-tuning. The key innovation lies in strategic data processing by modular agent orchestration that achieves competitive performance without the resource overhead of supervised fine-tuning (Johnson and Lee, 2025).

Distinguishing Characteristics:

- **Our Approach vs. Fine-tuned Models** (BART, T5+QLoRA, Legal-LED): We eliminate training overhead (120-200 GPU hours, 7,000+ annotations) (Jain and Kumar, 2024; Sharma and Reddy, 2024) while achieving competitive metrics. Our modular design enables rapid adaptation to new case types through prompt modification rather than retraining.

- **Our Approach vs. Direct LLM Prompting** (GPT-4, Llama 2-chat): We decompose summarization into specialized subtasks where zero-shot prompting excels, rather than expecting single-step generation.

- **Our Approach vs. General Agentic Frameworks** (ReAct, HuggingGPT): We design domain-specific workflows optimized for legal document structure (Yao et al., 2023b; Shen et al., 2023). While ReAct uses dynamic action selection and HuggingGPT employs runtime task planning, our fixed pipelines prioritize transparency and predictability for legal applications.

- **Our Approach vs. Contract Analysis Systems** (CUAD, ContractNLI): We address judgment summarization, which requires synthesizing multi-party arguments, chronological reasoning, and abstractive narrative generation, challenges distinct from contract clause extraction (Hendrycks et al., 2021; Koreeda and Manning, 2021).

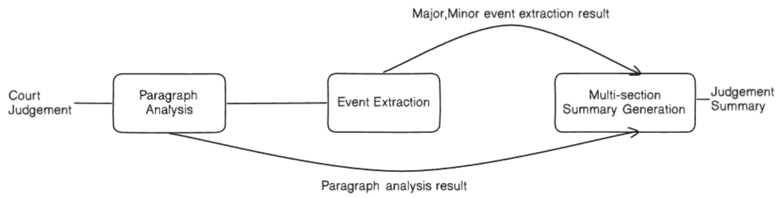
This positioning separates our work from traditional fine-tuned approaches while utilizing the flexibility and interpretability of agentic architectures. To our knowledge, this is the first application of modular, zero-shot agentic workflows to full-length Indian judicial decisions, demonstrating that strategic task decomposition can rival resource-intensive supervised training. By breaking down legal summarization into specialized processing stages, our framework operates at the task-routing level, selecting appropriate processing strategies for different document components while maintaining zero-shot generalization capability.

3 Methodology

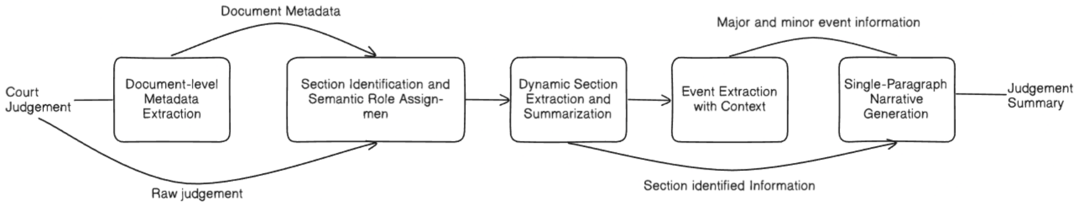
This section details our methodology for legal judgment summarization, describing both proposed modular agentic workflows designed for zero-shot operation.

3.1 Architectural Principles and Implementation Framework

Our architectural design prioritizes reliability, scalability, and maintainability by structuring workflows as a sequence of specialized, independent processing stages that produce intermediate outputs like paragraph classifications and event timelines, thereby creating a durable audit trail for traceability. This modular independence facilitates isolated validation and debugging, while the system's implementation relies on a purely zero-shot methodology orchestrated via the LangGraph framework. Utilizing Google Gemini 2.5 Flash as the primary backend due to its 1M token context window, native JSON support, and cost-effectiveness, the framework avoids task-specific training and remains compatible with other large-context instruction-tuned LLMs such as OpenAI GPT-4 and Claude.



(a) Lexical Modular Summarizer (LexA) (3-stage)



(b) Semantic Agentic Summarizer (SemA) (5-stage)

Figure 1: Architectures of the two proposed agentic summarization workflows.

3.2 The Lexical Modular Summarizer (LexA)

LexA is a three-stage modular architecture (see Figure 1a). This workflow type is categorized as a Level 2 Router Workflow with a hybrid extractive-abstractive strategy, having an extractive bias in early stages. Each processing stage results in an intermediate output with file-based storage at each stage.

1. **Paragraph Analysis:** This stage pre-processes input documents using regex patterns to extract numbered paragraphs using numbered paragraph markers (e.g., "1.", "2.", "3.") commonly found in Indian judgments. For each paragraph, we invoke the LLM with a classification prompt to identify segments as facts, legal issues, arguments, court reasoning, rulings, or procedural history. The process also involves extracting comprehensive metadata, including key topics, a legal significance score, the parties involved, relevant dates, specific legal concepts, and citations. All this information is compiled and exported as structured JSON, featuring detailed paragraph-level annotations. This paragraph-level classification enables downstream stages to focus on legally significant content while preserving exact phrasing for lexical overlap metrics.

2. **Event Extraction:** This stage identifies both major and minor legal events and constructs a chronological timeline to preserve key factual n-grams (dates, procedural terms) that contribute to ROUGE scores while providing narrative structure. Major events include case filings, judgments, appeals, and significant motions; minor procedural events include notices, adjournments, and document submissions. For each paragraph in the JSON file, we invoke the LLM with an event disambiguation and temporal ordering prompt. The final output is a structured event timeline as a JSON file that details the relationships between these events.

3. **Multi-section Summary Generation:** Reference summaries of our benchmark datasets are multi-section. We chose this format to maximize ROUGE overlap. We select high-significance paragraphs and all major events to invoke the LLM with a synthesis prompt, emphasizing n-gram preservation. Explicit instruction to preserve terminology maintains lexical fidelity. This process generates distinct sections, including an executive summary, factual background, legal issues, an event timeline, court reasoning, and the decision. The target length for the summary is determined by matching the 25–30% compression ratio.

3.3 The Semantic Agentic Summarizer (SemA)

SemA is a five-stage integrated architecture (see Figure 1b) designed for BERTScore, which emphasizes semantic similarity and deep legal understanding. This workflow operates as a Level 2 Router Workflow with a hybrid extractive-abstractive strategy, having an abstractive bias in later stages.

1. **Document-level Metadata Extraction:** This stage extracts key case metadata, including case number, court, date, parties involved, and judges from the raw documents. All this information is compiled and exported as structured JSON. Document-level understanding provides context for semantic processing in subsequent stages. It then evaluates the document's overall quality and completeness. Finally, it analyzes the structure of the citation network, identifying connections to statutes, precedents, and regulations.

2. **Section Identification and Semantic Role Assignment:** This stage defines the structural components of the documents, such as sections and subsections (facts, legal issues, arguments, reasoning, and conclusion), using structural markers and semantic analysis, while carefully maintaining the original legal terminology and phrasing. It also identifies clear section boundaries and assigns specific semantic roles to the different content blocks. All this information is compiled and exported as structured JSON. Semantic role identification enables abstractive generation that preserves legal reasoning structure rather than surface form.

3. **Dynamic Section Extraction and Summarization:** This stage extracts paragraphs for each identified section using granular classification and then groups them into sections based on their estimated types. It then generates section-level summaries using abstractive prompting and builds a hierarchical document structure. Section-level abstraction enables semantic compression while maintaining argumentative coherence.

4. **Event Extraction with Context:** This stage identifies a comprehensive event timeline and scores the legal significance of each event on a 1–10 scale based upon the document structure and context (which section, which argument does each event support?). It also analyzes event relationships (causal chains) and integrates these events with the structural layout of the argument. This context-aware event extraction supports semantic coherence in the final narrative by linking events to their legal significance.

5. **Single-Paragraph Narrative Generation:** This stage takes as input all the intermediate outputs stored in the workflow memory (analysis, sections, structure, and events) and invokes the LLM with a semantic synthesis prompt. This generates a single paragraph summary as a coherent narrative, targeting a length of 150–300 words. The single-paragraph format encourages semantic synthesis and conceptual abstraction. The focus on "reasoning chains" and "legal narrative" aligns with BERTScore's semantic similarity measurement.

4 Experimental Setup

To ensure a systematic and reproducible evaluation, we followed a structured experimental procedure encompassing both automated and expert-based assessments. Each test document goes through standardized preprocessing, followed by sequential execution of our proposed workflows. Intermediate results were stored for diagnostic analysis, enabling detailed error tracking and interpretability checks.

4.1 Baseline Models and State-of-the-Art Approaches

To assess the effectiveness of our proposed agentic workflows, LexA and SemA, we systematically compare them against extractive baselines, fine-tuned large transformers, and zero-shot or few-shot LLM baselines. We evaluated strong open-source and commercial models, including Llama 2-chat (70B, 7B), Gemini 2.5 Flash, and GPT-4, using standardized prompts; specifically, Gemini 2.5 Flash and GPT-4 served as zero-shot baselines to establish the direct performance limit of raw LLM API usage.

4.2 Datasets and Evaluation Metrics

4.2.1 Benchmark Datasets

Our workflows are evaluated on the two standard Indian legal datasets which are **CivilSum** and **IN-Abs**

Test Set Composition: We created a test set of a total of 50 Indian Court judgments, including 25 randomly selected documents from each dataset (CivilSum, IN-Abs), which we call Sum-IPL. It contains two parts: Sum-IPL-CivilSum and Sum-IPL-IN-Abs. We evaluate our workflows on Sum-IPL.

4.2.2 Quantitative Evaluation

For our quantitative evaluation, we employed a suite of automatic metrics to ensure an objective comparison. Specifically, we utilized ROUGE-1, ROUGE-2, and ROUGE-L (via py-rouge v1.1) alongside BERTScore (via bert-score v0.3.11) to assess text quality. Additionally, we tracked practical efficiency and accuracy metrics, including API cost per document, average processing time in seconds, and event coverage to measure the capture of key reference events.

4.2.3 Qualitative Evaluation

Three final-year LLB stusxdents from Kurukshetra University, India, were employed as experts to assess the human-aligned quality of the summary, beyond automatic metrics. Each of these three evaluators independently rated 10 randomly selected outputs (five each from the Sum-IPL-CivilSum and Sum-IPL-IN-Abs test sets) on six key criteria: factual accuracy, legal precision, completeness, coherence, conciseness, and overall quality, using a 1–10 scale. The document selection process ensured that each law student evaluated unique documents, resulting in a total of 30 documents evaluated from Sum-IPL during human evaluation.

The criteria assessed are **Factual Accuracy:** Correctness of facts, dates, parties, and events, **Legal Precision:** Appropriate use of legal terminology, concepts, and reasoning, **Completeness:** Coverage of essential case elements and event coverage, **Coherence:** Logical flow and readability, **Conciseness:** Appropriate length without redundancy, **Overall Quality:** Holistic assessment of summary quality

5 Results

5.1 Quantitative Performance

We attribute Llama 2-chat-70B's lower performance (0.374 ROUGE-1) to its few-shot prompting approach without task decomposition, as reported in prior work (Malik et al., 2024). Our modular workflows demonstrate that strategic task decomposition can substantially improve zero-shot performance over direct LLM prompting.

5.2 Qualitative Expert Evaluation

Table 3 presents the evaluation results, where a Fleiss' kappa of 0.68 indicates substantial inter-annotator agreement on the defined criteria. While both LexA and SemA achieve strong, comparable quality scores (7.5–8.5 range), they remain statistically distinguishable from human-written summaries (9.0 range). Significant gaps persist in Completeness, Legal Precision, and Overall Quality, suggesting that while the automated workflows offer practical utility, they still struggle to match the nuance and comprehensive coverage provided by human experts.

Qualitatively, experts praised both workflows for their accurate chronological structure and use of legal terminology, though they noted specific areas for improvement such as missed legal subtleties, incomplete citations, and occasional verbosity in LexA. The automated systems sometimes oversimplify complex arguments or commit minor factual errors. Overall, the workflows present distinct advantages: LexA excels in structured access and detail preservation, whereas SemA provides superior conciseness, narrative coherence, and conceptual clarity.

5.3 Error Analysis

Six types of failure patterns were identified in 22 out of 50 cases (25 from the CivilSum test set, 25 from the IN-Abs test set).

Complex Multi-Party Cases (6/50 cases): A specific challenge arises in cases involving more than 5 parties. The primary impact of this complexity is potential confusion in party roles and the risk of missing secondary legal issues. For example, in a corporate dispute with 7 parties, SemA misattributed an argument to the wrong party; meanwhile, LexA addresses this with enhanced party tracking, which includes explicit party-argument mapping during the paragraph analysis phase.

6 Discussion

The results demonstrate that modular, zero-shot agentic workflows can achieve competitive performance on legal document summarization without fine-tuning. Our findings offer significant implications for **resource efficiency**, as eliminating the need for GPU-intensive fine-tuning makes advanced legal AI accessible to resource-limited systems. The **modular design** ensures rapid adaptability across domains through prompt modification

Table 1: Comprehensive Performance Comparison Across Multiple Indian Legal Datasets. ROUGE-1 (R-1), ROUGE-2 (R-2), ROUGE-L (R-L), and BERTScore (BS) F1 scores are reported. All metrics are F1 scores.

Model/Approach	Dataset	R-1	R-2	R-L	BS	Fine-tune
Our Approaches - Zero-shot Agentic Workflows						
Lexical Modular Summarizer (LexA)	Sum-IPL-CivilSum	**0.6326**	0.4563	0.4508	**0.8902**	No
Semantic Agentic Summarizer (SemA)	Sum-IPL-CivilSum	0.4119	0.1253	0.2060	0.8474	No
Lexical Modular Summarizer (LexA)	Sum-IPL-IN-Abs	0.1951	0.0976	0.0928	**0.8299**	No
Semantic Agentic Summarizer (SemA)	Sum-IPL-IN-Abs	0.2014	0.0774	0.1104	**0.8122**	No
Zero-shot and Few-shot LLM Baselines						
Llama 2-chat-70B (Malik et al., 2024)	CivilSum	**0.374**	0.126	0.257	0.851	No
Llama 2-chat-7B (Malik et al., 2024)	CivilSum	0.371	0.126	0.254	0.851	No
Gemini 2.5 Flash (simple prompt)	CivilSum	0.389	0.132	0.184	0.782	No
GPT-4 (simple prompt)	CivilSum	0.412	0.148	0.195	0.804	No
Extractive Baselines						
Oracle Paragraph Extraction (Malik et al., 2024)	CivilSum	**0.331**	0.101	0.220	0.840	No
Random Extraction	CivilSum	0.198	0.042	0.165	0.712	No
Fine-tuned Transformer Models (For Reference)						
BART fine-tuned (Bhattacharya et al., 2019)	IN-Abs	0.450	0.180	0.230	0.820	Yes
T5 + QLoRA (Kumar and Sharma, 2022)	ILC	0.464	—	—	—	Yes
Legal-LED (Kapoor et al., 2024)	IL-TUR	—	—	0.330	0.860	Yes

Table 2: Comparison of Proposed Workflows on Various Aspects

Category	Aspect	LexA	SemA
Architectural Parameters	Workflow Type	Level 2 Router	Level 2 Router
	Summarization Strategy	Hybrid (Extractive-biased)	Hybrid (Abstractive-biased)
	Processing Stages	3-Stage Modular	5-Stage Integrated
	State Management	File-based	Memory-based
	Optimization Target	ROUGE metrics	BERTScore (semantic)
Performance Metrics (CivilSum)	ROUGE-1	**0.6326**	0.4119
	ROUGE-2	**0.4563**	0.1253
	ROUGE-L	**0.4508**	0.2060
	BERTScore	**0.8902**	0.8474
Performance Metrics (IN-Abs)	ROUGE-1	0.1951	0.2014
	ROUGE-2	0.0976	0.0774
	ROUGE-L	0.0928	0.1104
	BERTScore	**0.8299**	0.8122
Operational Characteristics	Processing Speed	Fast (45–60s)	Moderate (60–80s)
	Production Readiness	High	Medium (Research)
	Maintainability	High (Modular)	Medium (Integrated)
	Error Isolation	Excellent	Good
	Interpretability	Very High	High
Best Use Cases	Use Case	Production, high-volume; ROUGE	Research, semantic quality
Processing Efficiency	Avg. Processing Time	52 sec	71 sec
	Median Processing Time	48 sec	65 sec
	Range	28–145 sec	42–198 sec
	API Cost per Document	$0.03	$0.04
	GPU Hours Required	0	0
	Training Data Required	0	0

Table 3: Qualitative Evaluation (Scale: 1–10). Scores represent the average across evaluators. Inter-annotator agreement (Fleiss' kappa): 0.68 (substantial agreement).

Criterion	LexA	SemA	Human
Factual Accuracy	8.5	8.2	9.1
Legal Precision	8.3	7.8	9.3
Completeness	8.1	7.5	8.9
Coherence	7.9	8.1	9.0
Conciseness	7.6	8.4	8.7
Overall Quality	8.1	8.0	9.0

rather than retraining, while the explicit workflow guarantees **transparency** and provides a critical audit trail. Furthermore, the LexA and SemA models demonstrate that different optimization objectives can be achieved purely through architectural variation.

7 Conclusion

This paper demonstrates that modular, zero-shot agentic workflows can achieve competitive performance on legal document summarization without resource-intensive fine-tuning. We introduced two complementary architectures, Lexical Modular Summarizer (LexA) and Semantic Agentic Summarizer (SemA), that decompose summarization into specialized subtasks orchestrated through LangGraph. On the Sum-IPL-CivilSum test set, LexA achieves ROUGE-1 F1 of 0.6326 and BERTScore F1 of 0.8902, comparable to fine-tuned transformer models. Our findings suggest that strategic task decomposition and modular design can rival resource-intensive supervised training.

Future work should extend this approach to other languages, larger document collections, and explore integration with human-in-the-loop feedback mechanisms for continuous improvement. We also plan to investigate more sophisticated architectural patterns for handling complex multi-party cases and subtle legal reasoning.

Limitations

This work has several important limitations that should be considered:

1. **Language Scope:** Our evaluation focuses exclusively on English-language Indian Court judgments. The generalization to other languages or legal systems with different structural conventions remains untested.
2. **Dataset Scale:** Our evaluation uses a test set of only 50 documents. While this allows for detailed qualitative analysis, larger-scale evaluation would strengthen claims of generalizability.
3. **LLM Dependency:** Both workflows depend on the availability and performance of commercial LLM APIs. Changes in model capabilities, availability, or pricing could affect reproducibility and deployment feasibility.
4. **Expert Evaluation Scope:** Qualitative evaluation was conducted by three LLB students, not practicing lawyers. While their domain knowledge is sufficient for this assessment, evaluation by experienced legal practitioners would provide additional validation.
5. **Comparison Limitations:** Direct comparison with fine-tuned models (BART, T5+QLoRA) was not possible due to different evaluation datasets. Comparisons are primarily with zero-shot baselines and our own architectures.
6. **Complex Legal Reasoning:** The workflows struggle with highly complex multi-party cases and subtle legal distinctions, as revealed by error analysis and qualitative evaluation.

Ethics Statement

This work focuses on automating legal document summarization, which has important ethical implications. While automation can improve access to legal information for underserved populations, it also introduces risks of bias, hallucination, and misrepresentation of legal arguments. Our framework maintains interpretability through explicit workflow decomposition, enabling human review and oversight. We strongly recommend that any deployment of this system in real legal settings includes human-expert validation, particularly for cases with high legal stakes. Additionally, the use of LLMs in legal contexts raises privacy concerns regarding document handling by external API providers. Organizations deploying this system should implement appropriate data governance and privacy measures.

Acknowledgements

We thank the law students who participated in the qualitative evaluation and provided valuable feedback on summary quality. We also acknowledge the creators of the CivilSum and IN-Abs datasets, which formed the basis for our evaluation. This research was supported by National Institute of Technology Kurukshetra.

References

Paheli Bhattacharya, Kripabandhu Ghosh, Saptarshi Ghosh, Arindam Pal, et al. 2019. Legal case document summarization: Extractive and abstractive methods and their evaluation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*.

David Brown and Susan Taylor. 2022. Extractive methods for legal document summarization: A review. *Legal AI Quarterly*, 12(4):567–589.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. Legal-bert: The mismatch between domain-specific pretraining and downstream fine-tuning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2898–2904.

Michael Chen and Sarah Zhang. 2024. What are agentic workflows? patterns, use cases, and implementations. *AI Systems Journal*, 8(2):89–112.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Xue, Andrew M Huang, Dmitry Lepikhin, Yuanzhong Xu, Andrew M Dai, Zhifeng Chen, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Günes Erkan and Dragomir R Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of artificial intelligence research*, 22:457–479.

Neel Guha, Daniel E Ho, Julian Nyrup, Elizabeth Alexander, Andrew D Macey, and Divya Tenney. 2023. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models. In *Advances in Neural Information Processing Systems*.

Priya Gupta and Rakesh Sharma. 2024. Abstractive text summarization: State of the art, challenges and improvements. *Computer Science Review*, 42:100456.

Dan Hendrycks, Collin Burns, Spencer Chen, Spencer Ball, Frank Basart, Mason DeLucia, Jacob Li, and Radu Soricut. 2021. Cuad: An expert-annotated nlp dataset for legal contract review. In *NeurIPS Datasets and Benchmarks Track*.

Sirui Hong, Mingchen Zheng, Chen Jonathan, Alaa Faqih, Ron Arkin, and Carlos A Gomez-Uribe. 2024. Metagpt: Meta programming for multi-agent collaborative framework. In *Proceedings of the 12th International Conference on Learning Representations*.

Anshika Jain and Sriparna Saha. 2025. A comprehensive survey on legal summarization. *arXiv preprint arXiv:2501.17830*.

Deepak Jain and Sandeep Kumar. 2024. Fine-tuning bart for legal document summarization. In *Workshop on Legal Text Analytics*, pages 45–56.

Robert Johnson and Emily Lee. 2025. What is agentic ai? definition and differentiators in 2025. *Artificial Intelligence Review*, 45:1234–1256.

Abhinav Kapoor, Mohit Goyal, Anand Kumar, et al. 2024. Il-tur: Benchmark for indian legal text understanding and reasoning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, pages 1234–1245.

Yuta Koreeda and Christopher D Manning. 2021. Contractnli: A dataset for document-level natural language inference for contracts. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5806–5825.

Anand Kumar and Saurabh Sharma. 2022. Indian legal corpus (ilc): A dataset for abstractive summarization. *arXiv preprint arXiv:2210.10398*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Guohao Li, Hasan Abed Al Kader Hammer, Wenhao Yong, Hao Zhong, and Roberto Togneri. 2023. Camel: Communicative agents for 'mind' exploration of large scale language model society. In *Advances in Neural Information Processing Systems*.

Manuj Malik, Rohan Bhambhoria, Adam Roegiest, and Suzan Verberne. 2024. Civilsum: A dataset for abstractive summarization of indian court decisions. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2457–2467.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411.

Neha Patel and Rajesh Singh. 2024. Fusion of extractive and abstractive text summarization techniques for legal documents. In *International Conference on Natural Language Processing*, pages 234–245.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Priya Sharma and Karthik Reddy. 2024. T5 with qlora for efficient legal summarization. *Indian Journal of AI and Law*, 6(1):78–92.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. 2023. Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face. In *Advances in Neural Information Processing Systems*.

Noah Shinn, Alfredo Cassirer, Ashwin Bhandwaldar, Natasha Jaques, Justin Tan, Karthik Jakkam, and Carolina Parada. 2023. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*.

John Smith and Li Wang. 2024. Extractive and abstractive summarization techniques: A comprehensive overview. *Natural Language Processing Review*, 15(3):145–178.

Supreme Court of India. 2024. Structure and organization of indian judiciary. Official website.

LangChain Team. 2024. Langgraph: Framework for building stateful multi-agent applications. GitHub repository.

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Jingning Yang, Jiakai Zhang, Zhuosheng Chen, Jiawei Xie, Yaliang Huang, Dawei Song, et al. 2023. A survey on large language model based autonomous agents. *arXiv preprint arXiv:2308.11432*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Ichien, Ed H Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, pages 24824–24837.

Amanda Wilson and Vijay Kumar. 2024. Addressing ai bias and fairness: Challenges, implications and strategies for ethical ai. *Ethics in AI*, 18(2):234–267.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L Griffiths, Yuandong Cao, and Ankur P Parikh. 2023a. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems*.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Ankur P Parikh, and Hongkuk Jiang. 2023b. React: Synergizing reasoning and acting in language models. In *Proceedings of the 11th International Conference on Learning Representations*.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.

A Appendix: Detailed Prompts and Implementation Details

A.1 Paragraph Analysis Prompt

For each paragraph in the input document, the following prompt is used to classify the paragraph type and extract metadata:

`“Classify the following legal paragraph and extract metadata. Paragraph types: Facts, Legal Issues, Arguments, Court Reasoning, Rulings, Procedural History. Extract: key topics, legal significance (1-10), parties, dates, legal concepts, citations.”`

A.2 Event Extraction Prompt

For event extraction and temporal ordering:

`“Extract major and minor legal events from this paragraph. Major events: filings, judgments, appeals, motions. Minor events: notices, adjournments, submissions. Create a chronological timeline with event relationships.”`

A.3 Final Summary Prompt

For multi-section summary generation:

`“Generate a legal summary with sections: Executive Summary, Factual Background, Legal Issues, Event Timeline, Court Reasoning, Decision. Preserve n-grams from source. Target 25-30% compression ratio.”`

A.4 Semantic Role Assignment Prompt

For SemA semantic role assignment:

`“Analyze document structure and assign semantic roles to sections: Facts, Legal Issues, Arguments, Reasoning, Conclusion. Maintain legal terminology. Identify section boundaries.”`

A.5 Data Format Examples

Example output of LexA Stage 1 (paragraph_analysis.json):

```
{
  "paragraphs": [
    {
      "id": 1,
      "type": "Procedural History",
      "content": "The case was filed...",
```

```
      "metadata": {
        "topics": ["case filing", "
            jurisdiction"],
        "significance": 8,
        "parties": ["Appellant", "
            Respondent"],
        "dates": ["2020-01-15"],
        "concepts": ["writ petition"]
      }
    }
  ]
}
```


Example output of LexA Stage 2 (event_timeline.json):


```
{
  "timeline": [
    {
      "date": "2020-01-15",
      "event": "Case filed",
      "type": "major",
      "significance": 9
    }
  ],
  "relationships": [
    {
      "event1": "Case filed",
      "event2": "Hearing scheduled",
      "type": "follows"
    }
  ]
}
```


Example output of SemA Stage 1 (metadata.json):


```
{
  "case_number": "2020/SC/12345",
  "court": "Supreme Court of India",
  "date": "2023-06-15",
  "parties": ["Appellant", "Respondent
      "],
  "judges": ["Justice A", "Justice B"]
}
```


Example output of SemA Stage 2 (sections.json):


```
{
  "sections": [
    {
      "name": "Facts",
      "role": "factual_context",
      "paragraphs": [1, 2, 3],
      "boundary_markers": ["Para 1", "
          Para 5"]
    }
  ]
}
```

LLM Driven Legal Text Analytics: A Case Study For Food Safety Violation Cases

Suyog Joshi[1], Soumyajit Basu[1], Lipika Dey[1], Partha Pratim Das[1]

[1]Ashoka University, Sonepat, India

Correspondence: suyog.joshi@alumni.ashoka.edu.in, soumyajit.basu@alumni.ashoka.edu.in
lipika.dey@ashoka.edu.in, ppd@ashoka.edu.in

Abstract

Despite comprehensive food safety regulations worldwide, violations continue to pose significant public health challenges. This paper presents an LLM-driven pipeline for analyzing legal texts to identify structural and procedural gaps in food safety enforcement. We develop an end-to-end system that leverages Large Language Models to extract structured entities from legal judgments, construct statute-and-provision-level knowledge graphs, and perform semantic clustering of cases. Applying our approach to 782 Indian food safety violation cases filed between 2022-2024, we uncover critical insights: 96% of cases were filed by individuals and organizations against state authorities, with 60% resulting in decisions favoring appellants. Through automated clustering and analysis, we identify major procedural lapses including unclear jurisdictional boundaries between enforcement agencies, insufficient evidence collection, and ambiguous penalty guidelines. Our findings reveal concrete weaknesses in current enforcement practices and demonstrate the practical value of LLMs for legal analysis at scale.

1 Introduction

In 2024, the World Health Organization (WHO) reported that unsafe food causes around 600 million cases of foodborne diseases, and $420,000$ deaths annually. Despite the presence of comprehensive food regulatory acts across countries, food safety violations continue to pose a significant challenge to global public health. The complexity of food supply chains coupled with evolving legal frameworks, often results in inconsistent enforcement and delayed policy response. Though the exact statistics are not available for India, research articles report that among several other developing nations in South East Asia, food adulteration is widespread in India[1]. The Food Safety and Standards Act of India was passed in 2006 and thereafter, various provisions of the Act have to force through several notifications. Despite this, food safety violations remain a national concern. According to a recent article published by National Law Institute University, the Food Safety and Standards Authority of India (FSSAI), responsible for regulating and overseeing food safety in India, faces several challenges, including limitations of their regulatory purview, infrastructure deficiencies, limited scope of regulations, etc.[2]. In a study conducted by Shukla et al. (2014) in 2014, it was emphasized that when it comes to food safety assessment and implementation of quality, there exists a gap in infrastructure and risk-based approach in both implementation and enforcement. While this could possibly explain why food safety violation remains critically high, ours is an evidence-based approach to objectively unearth these gap areas through analysis of court cases. Our investigation revealed some interesting insights. While the number of court cases are not overwhelming, our study reveals that most of these were filed by those accused of violation against the authorities, and a large number of them are also won by the appellants. This prompted us to dive deeper into the case files to gain insights about what is happening and identify the possible loopholes, wherever they are.

Legal text analytics plays a critical role in identifying gaps, performing causal analysis, and thereby contributing towards strengthening regulatory oversight. By automatically analyzing statutes, compliance guidelines, inspection reports, and judicial rulings, legal text analytics systems can identify

[1]https://ncdc.mohfw.gov.in/wp-content/uploads/2024/04/Food-borne-Diseases-and-Food-Safety-in-India.pdf

[2]https://nliulawreview.nliu.ac.in/blog/fssais-regulatory-apathy-and-indias-marginal-consumers-a-case-for-decentralized-food-safety/

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 41–52
December 24, 2025 ©2025 Association for Computational Linguistics

emerging patterns of non-compliance, inconsistencies across regional laws, and gaps between regulations and enforcement practices. This work was initiated to obtain possible insights about the food safety violation landscape, the type of cases reported, the food items or malpractices frequently associated, and most importantly the possible reasons for the practice not reducing, despite regular actions by FSSAI. The intent was to analyze the case proceedings and judgments in the context of the Food Safety and Standards Act of India, 2006, to identify possible causal factors. In this paper we propose an evidence-based approach that uses Natural Language Processing techniques to identify the structural and procedural factors that could be influencing the case volumes in the judicial system.

One of the key bottlenecks faced while designing legal text analytics systems earlier was the lack of annotated corpora to train the systems. This has been substantially eased by the Large Language Models (LLMs), which have already been exposed to fair amounts of legal corpora from across the world. LLMs can play a crucial role in legal text analytics by summarizing regulatory provisions and case documents, extracting obligations, penalties and other relevant key entities, and also in answering questions about how the judicial reasoning progressed. Together, these can provide deeper insight into the legal landscape associated with food safety in India. The insights can enable regulators, policymakers, and researchers to gain a data-driven understanding of legal landscapes, facilitating proactive interventions and harmonization of global food safety standards. Ultimately, the integration of legal text analytics into food governance frameworks can enhance transparency, improve compliance monitoring, and contribute to safer and more accountable food systems around the world.

This paper presents how Large Language Models (LLMs) and knowledge graphs can be used for legal text analytics. The research develops an end-to-end pipeline that leverages LLMs to extract structured entities and factual context from legal judgments, constructs a statute-and-provision-level knowledge graph, and aligns India's Food Safety and Standards Act, 2006 with equivalent laws in Finland and the United Kingdom using a hybrid semantic similarity and LLM-based matching process.

2 Review of Related Work

Natural language processing of legal texts poses unique challenges due to their lengths, denseness and use of specialized vocabulary. However their analysis is important to understand the effectiveness of the legal system. While manual processing is time-consuming and prone to error, automated analysis is difficult due to the complexity of syntax, archaic jargon, and strict semantics. The text processing tasks traditionally included summarization, named entity recognition and structured information extraction. Ariai et al. (2024) present a comprehensive view of language processing tasks for legal text. The integration of Large Language Models (LLMs) in legal technology is rapidly transforming the landscape of legal research and document analysis, and several new applications like predicting missing citations, legal analytics are also gaining popularity with these models. The rapidly changing landscape of legal technologies centered around the use of LLMs are presented in many articles like Mayer (2023), Siino et al. (2025), Padiu et al. (2024), Ződi (2024). Named Entity Recognition (NER) in legal texts are designed to identify entities like statute names, case parties, courts, etc. Early approaches treated NER as a token classification task (Skylaki et al., 2020) but use of abbreviations, synonyms, acronyms make this task heavily error-prone. A recent survey evaluating approaches ranging from traditional rule-based systems to LLMs reports that `GPT-4` achieves superior performance on legal NER benchmarks, outperforming smaller models (Deußer et al., 2024). This suggests that LLMs can better use context to resolve ambiguities e.g. linking "Section 420 of FSS" to the correct law. NER based approaches have been also used to extract structured facts like case facts, charges, outcomes, etc. from judgments. Recent work like LegalLens reports the use of `GPT-4` to label violation entities in unstructured legal text (Hagag et al., 2024). In their work Deußer et al. (2024), also report that `GPT-4` outperformed smaller language models on legal NER and text classification tasks.

Use of LLMs have enhanced the quality of structured information extraction from legal texts, in turn increasing their use in legal text analytics. In a recent publication, Pereira et al. (2025) have shown the utility of using GPT-4o to extract factors for Brazilian consumer law judgments, without any further fine-tuning. LLMs have also been reported for immigration policy analysis by Brown (2025).

Li et al. (2025) propose LegalAgentBench, a comprehensive benchmark specifically designed to evaluate LLM Agents in the Chinese legal domain.

The quality of summarizing long legal documents like contract agreements also improved phenomenally with the use of LLMs. The work reported in Davenport (2025) proposed the use of LLMs for hierarchical segmentation of large documents in combination with chain-of-thought prompting and multi-stage summarization techniques. This article reports that OpenAI's API can not only summarize and analyze long contracts quite well, but also capture critical obligations and clauses with high accuracy and efficiency. Similar results were reported in Litaina et al. (2024), who presents results from an experiment in which they use GPT-3.5 and Gemini to analyze type and subject of contracts to obtain insights about involved named entities and their relationship(s). Handwritten contracts were first digitized into plain text format using an AI-powered tool called Transkribus. Their experiments demonstrated that the LLM-generated responses outperformed humans in precision, but not in recall. The performance of LLMs can be further improved by the use of legal Knowledge Graphs (KG). A legal knowledge graph typically represents cases, laws, and concepts as nodes, with edges for relationships like citations or involvement. In a different approach, the work done by Dhani et al. (2024) presents an Indian legal KG built with regulatory documents and case documents. The nodes include cases, judicial orders, statutes, parties involves, while edges capture citations, applicable statutes etc. A Graph Neural Network, is trained on the knowledge graph to predict similar cases and thereafter suggest missing citations. The KG-based approach is reported as yielding higher precision while retrieving similar cases. Using LLMs for legal reasoning is increasingly gaining traction. A framework for unified legal reasoning that combines rule-based, abductive, and case-based approaches, and then investigate possible methods for their integration with LLMs is proposed in Nguyen et al. (2025).

3 Knowledge Graph Design for Regulation Act and Case Documents

A knowledge graph was designed to capture some of the key information that is relevant for insightful analysis of cases, not just for the present work, but also to be used for other kinds of comparative analytics of legal documents. The design was based on concepts and relationships presented in Leone et al. (2019) enhanced by the main features of the provisions and statutes as followed by Indian legal framework. The statutes represent the broader issues being discussed. For example, a tax-related case may cite the Income Tax Act, while the provisions represent the specific clauses and subsections that provide the precise context for the case's arguments and judicial reasoning. Another portion of the knowledge graph contains details about individual cases. These include case-specific details like the judge overseeing the case, names of petitioners and respondents, the acts, statutes, section numbers cited in the case, the names of locations and organizations mentioned in the document, the final judgment in the case.

Figure 1 presents the design of the knowledge graph that stores country-specific regulatory information created from regulatory documents. This graph contains as nodes the names of different acts, statutes, provisions and their descriptions. These are connected to each other using the part-of relationship. Initially, a graphRAG based approach was also tried. However, with very little control over the knowledge graph that was created, it was difficult to perform analytical queries of the kinds that we wanted to.

On the right hand side of Figure 1, the knowledge graph design for individual cases is presented. Each case gets connected to those nodes of the law knowledge graph, that are cited in it. The prompt used for the extraction of different entities, their resolution and creating entries for the knowledge graph is given in A.2.

Figure 2 presents the full knowledge graph created from the Food Safety Regulatory document published in 2006. Connecting the cases to the statutes and provisions in the knowledge graph allows for meaningful insights to be extracted about the nature of the cases present in a corpus. The graph can be queried for quantitative insights. This includes information like the most commonly cited statutes across cases, clustering of cases based on citation similarity, and finding the cases most similar to a chosen case. These queries can provide information about patterns across the corpus, such as understanding the frequency of specific legal interpretations, or identifying the most common reasoning provided in specific types of cases.

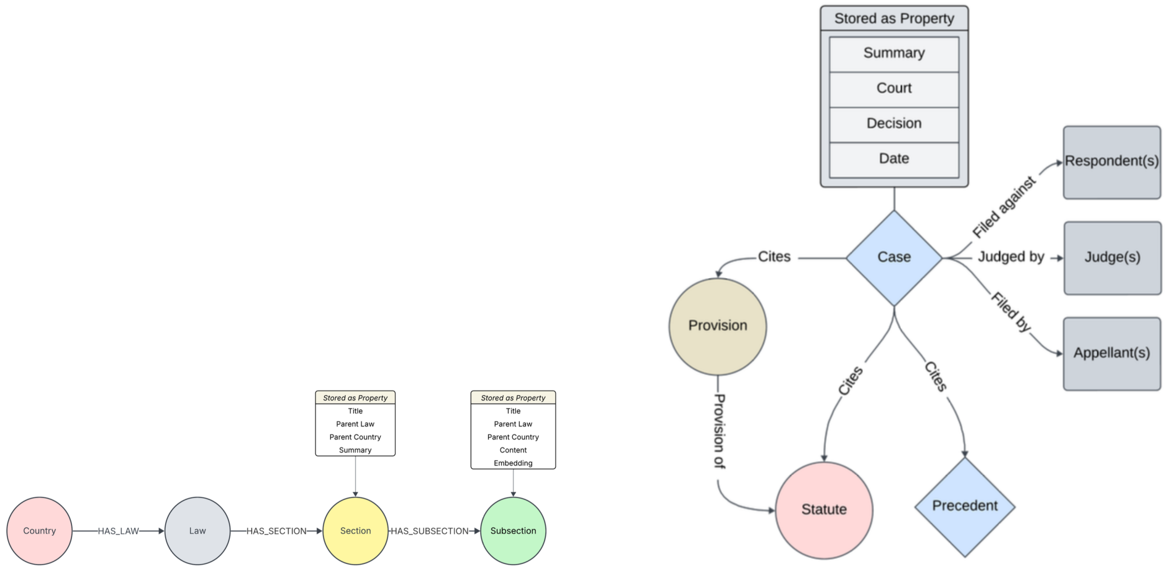


Figure 1: (left) Knowledge Graph Model for storing Regulatory information about statutes and sections (right) Knowledge graph for each case document

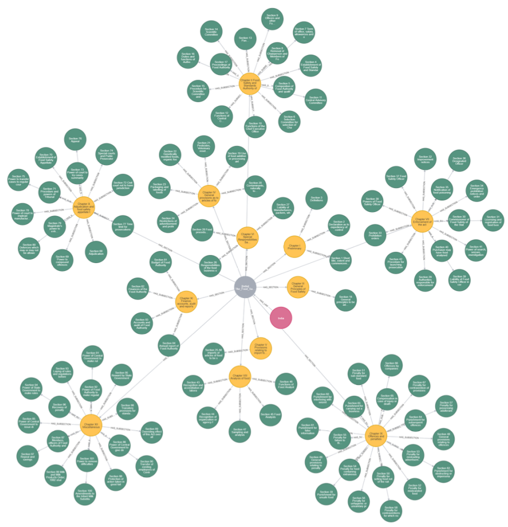


Figure 2: Knowledge Graph curated from Food Safety and Standards Act 2006, India

4 LLM Driven Pipeline to Obtain Insights from Case files

Figure 4 presents the pipeline that we have deployed for LLM-driven case document analysis. While the core activity is to extract different kinds of information that can help in analytics-driven insight generation, the content analytics pipeline consists of two different threads. Case documents were downloaded from a website called Indiakanoon.org using a custom-designed crawler. The pipeline consists of two independent threads, each of which processes each case document. The analytical activities utilizes the outputs of both. Thread 1 analyzes each document and creates a structured summary of the court proceedings. The prompt for creating the summary is given in A.3

Thread 2 parses each case to extract different types of entities that are required to populate the knowledge graph. The end result of the pipeline is a structured representation for the documents, their summaries and all the case-specific details that are extracted from the document. This consolidated list is now ready for further analysis to obtain insights.

5 Clustering Case Summaries, Decisions and Reasons Given for the Decisions

To discover the patterns across the food safety violation cases, we explored a semantic similarity based clustering methodology using the structured case summaries obtained earlier. Each case summary contains the case overview, key facts from the case, the decision and the judge's reasoning. Contextual embeddings of these summaries are created using different mechanisms. With the summaries now represented as vectors, we use the k-means algorithm to cluster the cases into meaningful groups. The optimal number of clusters is determined using the silhouette score and Davies-Bouldin (DB) index together. Specifically, we optimize $k^* = \arg\max_{k\in[2,20]} \frac{S(k)}{DB(k)}$, where $S(k)$ and $DB(k)$ are the silhouette score and the DB index,

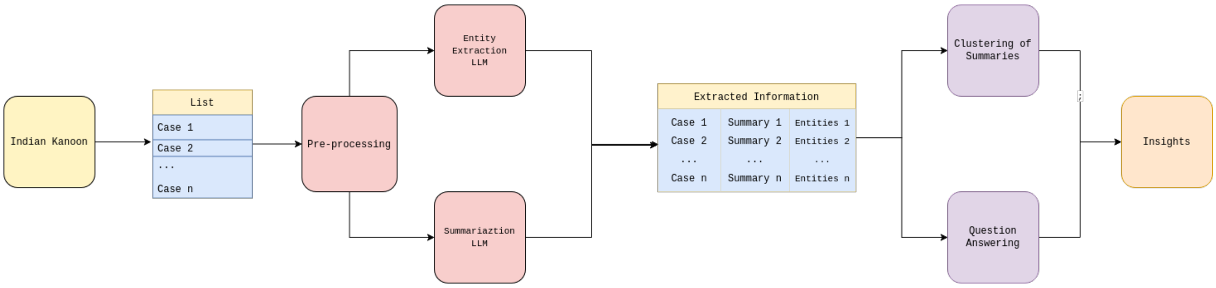


Figure 3: LLM-driven case document processing Pipeline - Information extraction from individual documents followed by analytics over the repository

respectively, for k clusters. Finally, each cluster is passed through an LLM to generate a descriptive labels for the cluster. In this paper, we report results for three different embedding model and LLM combinations. The first set of embeddings were created using OpenAI's `text-embedding-3-large` model, and the corresponding cluster labels were generated using `GPT-4o`. The second set used Google's `gemini-embedding-001` for embedding generation and `Gemini-2.5-Flash` for generating the cluster labels. A third set of results were obtained using Alibaba's `qwen3-embedding-8b` for embedding generation and `Qwen3-235B-A22B-Instruct-2507` for cluster labeling. The intent was to study the susceptibility of the analytical results to different embedding generation process. For a robust analysis, the case summaries should cluster more or less identically across all three embedding spaces, and also be assigned semantically similar labels. Results from Section 6.4 show that this indeed is the case.

6 Analysis of Food Safety Violation cases filed between 2022 - 2024

6.1 Data Collection

Court proceedings related to food safety were collected from indiankanoon.org using a webscraper built using Selenium and BeautifulSoup. Prior consent to use judgments available on the website was received from the IndianKanoon team. To ensure all cases mentioning "Food Safety" were included, a search term with a week-long window was created, which ensured that cases were downloaded one week at a time. This is because IndianKanoon limits search results to 400 cases, so limiting to one week ensures all cases from the period are downloaded. Judgements passed between January 1, 2022 and December 31, 2024 were downloaded. A total of 7233 cases were collected, out of which duplicates were dropped along with files containing incorrect formatting and files exceeding the `GPT-4o-mini` context length of $128,000$ tokens. The remaining 7040 files were selected for processing after minimal pre-processing. Links common to all documents, such as links pointing to IndianKanoon or judis.nic.in were removed. The remaining texts were passed downstream as-is for extracting the information about petitioners, defendants, court and the date of case filed. First level analysis revealed that a majority of them were related to tobacco products. Though tobacco itself is not considered as a food product under the Food Safety and Standards Act (FSS Act), its use is banned in edible items. This might have resulted in a high number of cases fetched during crawling and hence were dropped. A total of 782 unique cases finally remained which were used for downstream analysis.

6.2 Case Document Processing and Summarization

Each case document was passed through the summarization and information extraction pipelines implemented using `GPT-3.5`. The pipeline extracts structured information including petitioners, respondents, court details, case overview, key facts, legal issues and arguments, court's reasoning, and the final decision. A sample output demonstrating the extracted structure is provided in Appendix A.1.

To assess the faithfulness of the LLM-generated summaries to the original content, we computed the BERT and ROUGE-L scores between the generated summaries and the original case documents. An average BERT score of 0.624 represents a fair amount of semantic similarity between the summaries and their parent documents. Average ROUGE-L score of 0.275 represents structural similarity, that is represented by the longest common subsequences present in both the document and its summary. This

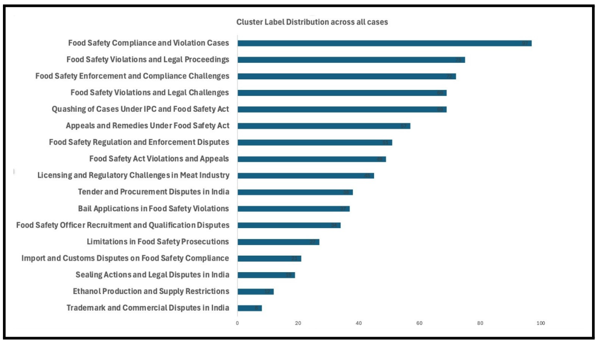


Figure 4: Distribution of cluster labels across all cases

part of the experiment could not be replicated due to resource constraints.

6.3 Results of Clustering of Food-safety related Cases Decided Between 2022 - 2024

The experiments were repeated many times. On an average, GPT obtained 17 clusters, Gemini based analysis yielded 18 and Qwen found the optimal number to be 16. Figure 5 presents the distribution of the cases across cluster labels for all the three alternatives, generated using t-SNE, which essentially produces two-dimensional plots for high-dimensional embeddings. The colours are randomly assigned by the visualizer for each plot, and has no other significance. In other words, blue dots across the plots are not necessarily associated to the same underlying documents across the different embedding spaces. Indeed, the plots do show that the case summaries cluster quite similarly across different embedding spaces, and the optimal numbers are also pretty close.

6.4 Comparing Cluster Labels for Robustness Analysis

The thematic alignment between the cluster labels generated by the three models was obtained using a semantic consensus metric based on Sentence-BERT (SBERT) embeddings for the cluster labels generated by the three LLMs. These embeddings were generated using `all-MiniLM-L6-v2` model, and their pairwise cosine similarities were computed to determine semantic equivalence. A similarity threshold of $\tau = 0.55$ was established to account for terminological variations, such as "Ethanol Production" versus "Ethephon and Ethanol Regulatory Disputes", while maintaining semantic precision. The analysis yielded a robust average consensus score of 90.3% across all model pairs. The highest alignment was observed between `Gemini-2.5-Flash` and `Qwen-3` (94.4%), followed by `GPT-4o` and `Gemini-2.5-Flash` (88.2%), and `GPT-4o` and `Qwen-3` (88.2%). Detailed pairwise similarity scores for all cluster label comparisons are provided in Appendix A.5.

A granular examination of the divergent labels (similarity < 0.55) revealed that the remaining discrepancies were largely structural rather than substantive. For instance, `GPT-4o` tended to isolate specific administrative actions like "Sealing Actions", while other models grouped these with broader regulatory enforcement themes. Figure 6 presents the agreement between the labels generated by different methods. The strong cross-model agreement on labels validates the robustness of our clustering approach and confirms that the identified legal themes are model-independent.

On manual inspection, GPT-generated labels were found to be most comprehendible as well as at the right granular levels. For example, the most frequent label *Quashing Food Safety and Allied Criminal Proceedings* generated by `GPT-4o`, links to three different labels generated by Gemini, namely *Food Safety Act: Limitation and Quashing Proceedings*, *Food Safety Procedural Violations and Quashing* and *Food Safety Act Violations and Quashment*. Case-wise label assignments were also reviewed manually for robustness check. Case document wise, it is found that 90% of the cases that are assigned to the cluster labeled *Quashing Food Safety and Allied Criminal Proceedings* by `GPT-4o` are assigned to the cluster labeled *Food Safety Act: Limitation and Quashing Proceedings* by Gemini. The next highest category of cases belong to the cluster labeled *Food Safety Compliance and Violation Cases*. 65% of these cases belong to the cluster labeled *Food Safety Act Procedural Violations* and the remaining 35% cases belonged to the cluster labeled *Food Safety Procedural Violations and Quashing*. After reviewing all the labels, for the final insight extraction, we decided to use the `GPT-4o` labels for clarity and distinctiveness. Figure 4 presents the distribution of the cluster labels generated by `GPT-4o`.

6.5 Insights Extracted

A detailed analysis of the 782 cases is now presented. Analysis of petitioners and respondents shows that 96% of these cases were filed by individuals and organizations against a representative

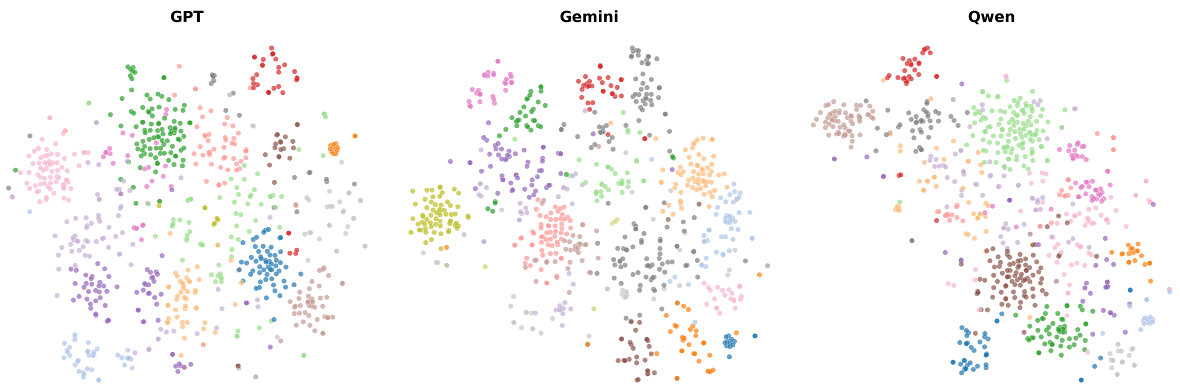


Figure 5: t-SNE Visualization of clusters generated by different embedding methods; distinct clusters are obtained for all methods showing the separability of the cases irrespective of the embedding space

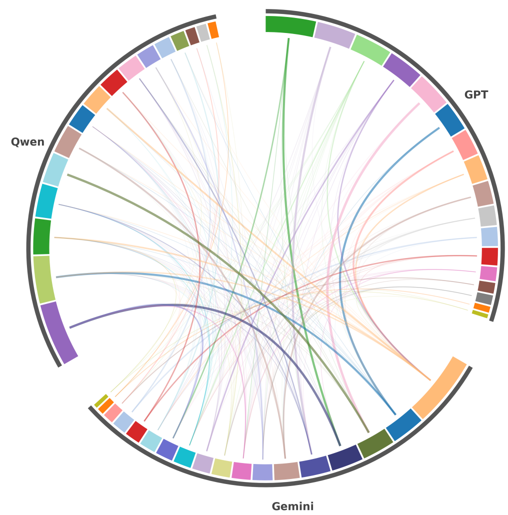


Figure 6: Agreement between the labels generated by different embedding methods. Gemini-generated labels are good proximities to the other two sets. Granularity of GPT generated labels appear to be best as the most frequent labels are more uniformly distributed. Each corresponds to one big and one small cluster of Gemini.

of the state authority. These cases were mostly filed challenging an action or a decision taken by a representative of a State agency. In a few cases, these were petition against government announcements. Based on the extracted information about in whose favour the judge's decision go, Figure 7 shows that 60% of the cases filed against state authorities were won by the appellants or petitioners. Assuming that the state authorities would have acted against an individual or organization only if a food safety violation act was detected, this statistics by itself is quite intriguing. It reveals that most of the appeals against the authority's actions were given a judgment which went in favour of the appellant or petitioner, and not the state authority. Figure 8 shows the distribution of the cluster labels obtained using `GPT-4o` for those cases which were won by individuals or organizations against state agency representatives. We opted for `GPT-4o` labels for the same reason as cited earlier.

To understand the situation better, we present the cluster label distributions for the above cases separately in . Further analysis of the top 5 labels, other than those for bail applications are given below:

- Quashing of Cases Under IPC and Food Safety Act: These were cases where the court quashed the FIR registered against the petitioners. For many cases, the court observed that the FSS Act allows only Food Safety Inspectors to initiate prosecution, and not Police or other authorities. Some were quashed since the action was taken after stipulated time-period from FSS violations were observed. Similarly, if a case was filed under provisions of both IPC and FSS, the court observed that the dual role of the complainant and investigating officer compromises on fair trial rights of the appellant.

- Food Safety Compliance and Violation Cases: This category majorly includes all cases in which the judge's decision went in favour of the appellants citing procedural lapses on part of the state authorities while conducting the necessary tests for establishing food safety violations

- Food Safety Enforcement and Compliance Challenges: In this category, the court most often found the respondents' arguments unacceptable. the details vary from case to case. For example, in a case titled "R Shanmughasundaram vs The Food Safety Officer on 21 September 2023", that involved a fine imposed for misbranding oil products, the petitioner challenged the fine amount of Rs. 3,00,000, which they found to be excessive. The state authorities argued that a sample of groundnut oil was actually found to be Palm Oil and there is a significant price difference between the two. Though there was merit in the case, the court noted that the District Revenue Officer did not fully consider the factors outlined in Section 49 of the Food Safety and Standards Act, 2006, when determining the fine. The petitioner was let off with an undertaking to prevent future misbranding, a smaller amount of fine and an order for the state authorities to vacate the entire premises without any qualifications.

- Food Safety Violations and Legal Proceedings: These were mostly cases which dragged on for a long time. In some cases, the petitioners pleaded not guilty to the crime as they were only resellers or employees. In some cases, though the crimes were acknowledged, since the petitioners had spent substantial time in imprisonment, the court decided to replace the sentence of imprisonment with penalties.

- Food Safety Regulation and Enforcement Disputes: Under this category, the court noted that the allegations of violations reported were not substantiated by sufficient evidence or test results that were acceptable.

Detailed analysis of all the clusters revealed that there were major procedural issues due to which many food safety violation cases were not being penalized sufficiently. It includes lack of knowledge or actual lack of concrete boundaries between the responsibilities on the part of the Food Safety officers and other law enforcement agencies like the police. Unclear rules about the penalties to be applied also allows petitioners to possibly get away with their crimes. In many cases, resellers or employees were penalized, which doesn't affect the root cause of safety violation, and could be also one of the reasons for the violations to continue.

The analysis clearly establishes the need for safeguarding the interests of public health through re-

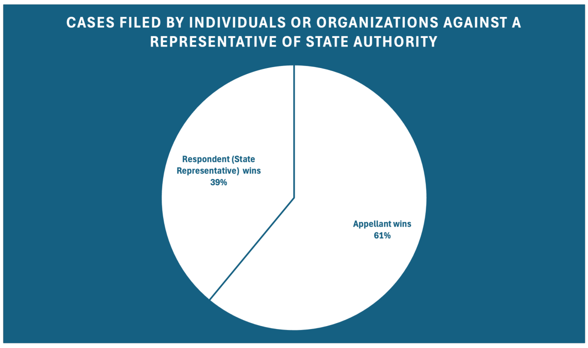


Figure 7: Distribution of judge's decision for cases filed by individuals or organizations against state authorities - most cases are lost by the State authorities

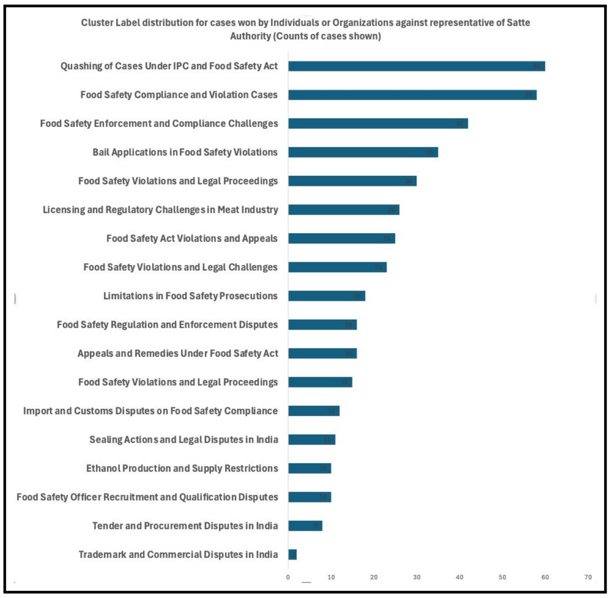


Figure 8: Distribution of cluster labels for cases won by individuals or organizations against state authorities

moval of these loopholes in the regulation and its implementation along with education and awareness among all citizens.

7 Conclusions and Future Work

In this work, we have demonstrated the potential of Large Language Models (LLMs) in establishing an evidential approach to understand the landscape of food safety violations by analyzing the food safety–related legal cases. This work has shown that by leveraging the advanced capabilities of LLMs in analyzing legal text, both information extraction and contextual reasoning can play significant roles in identifying the possible causes of recurrent violations. The insights that are revealed include the large number of cases that end up being quashed due to procedural errors and also an equally large number of them showing up the lacunae in the regulatory framework itself as well as enforcement challenged. The case study finds evidences in support of known issues that exist between the intentions of the regulation and implementation challenges. Lack of awareness among the authorities of law enforcement also surface as a key issue.

The research also initiated knowledge modeling for the regulations themselves. In future, we intend to continue our work towards comparative analysis of regulations from different countries, their implementations and the outcomes observed. The work will continue to focus on developing easily deployable explainable and domain-tuned LLMs that can integrate legal ontologies and cross-jurisdictional data to ensure trust and accountability. This area can contribute significantly towards strengthening of food safety systems across the globe.

Limitations

The proposed pipeline has been applied on food safety-related cases filed in Indian courts over three years from 2022 to 2024. The entire pipeline from initial document processing to clustering could not be repeated with multiple LLMS due to resource constraints. Only the clustering part was repeated for three different systems. Our future work would focus on employing the entire pipeline for larger set of case documents to obtain insights at multiple levels, including those related to food items and their propensity for safety violations. This would make the work more complete.

Acknowledgments

We are thankful to the Mphasis AI & Applied Tech Lab at Ashoka - a collaboration between Ashoka University and Mphasis Limited - for their support.

References

Farid Ariai, Joel Mackenzie, and Gianluca Demartini. 2024. Natural language processing for the legal domain: A survey of tasks, datasets, models, and challenges. *ACM Computing Surveys*.

Crystal Brown. 2025. *Leveraging generative AI and system dynamics for enhanced immigration policy analysis*. Ph.D. thesis, Worcester Polytechnic Institute.

Mark J Davenport. 2025. Enhancing legal document analysis with large language models: A structured approach to accuracy, context preservation, and risk mitigation. *Open Journal of Modern Linguistics*, 15(2):232–280.

Tobias Deußer, Cong Zhao, Lorenz Sparrenberg, Daniel Uedelhoven, Armin Berger, Maren Pielka, Lars Hillebrand, Christian Bauckhage, and Rafet Sifa. 2024. A comparative study of large language models for named entity recognition in the legal domain. In *2024 IEEE International Conference on Big Data (BigData)*, pages 4737–4742.

Jaspreet Singh Dhani, Ruchika Bhatt, Balaji Ganesan, Parikshet Sirohi, and Vasudha Bhatnagar. 2024. Similar cases recommendation using legal knowledge graphs. *Preprint*, arXiv:2107.04771.

Ben Hagag, Liav Harpaz, Gil Semo, Dor Bernsohn, Rohit Saha, Pashootan Vaezipoor, Kyryl Truskovskyi, and Gerasimos Spanakis. 2024. LegalLens shared task 2024: Legal violation identification in unstructured text. *Preprint*, arXiv:2410.12064.

Valentina Leone, Luigi Di Caro, and Serena Villata. 2019. Taking stock of legal ontologies: a feature-based comparative analysis. *Artificial Intelligence and Law*, 28(2):207–235.

Haitao Li, Junjie Chen, Jingli Yang, Qingyao Ai, Wei Jia, Youfeng Liu, Kai Lin, Yueyue Wu, Guozhi Yuan, Yiran Hu, Wuyue Wang, Yiqun Liu, and Minlie Huang. 2025. LegalAgentBench: Evaluating LLM agents in legal domain. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2322–2344, Vienna, Austria. Association for Computational Linguistics.

Tania Litaina, Andreas Soularidis, Georgios Bouchouras, Konstantinos Kotis, and Evangelia Kavakli. 2024. Towards LLM-based semantic analysis of historical legal documents. In *SemDH2024: First International Workshop of Semantic Digital Humanities, co-located with ESWC2024*.

T Mayer. 2023. AI and LLMs in legal technology: Revolutionizing research and document analysis. *Advances in Computer Sciences*, 6(1).

Ha Thanh Nguyen, Wachara Fungwacharakorn, May Myo Zin, Randy Goebel, Francesca Toni, Kostas Stathis, and Ken Satoh. 2025. LLMs for legal reasoning: A unified framework and future perspectives. *Computer Law & Security Review*, 58:106165.

Bogdan Padiu, Radu Iacob, Traian Rebedea, and Mihai Dascalu. 2024. To what extent have LLMs reshaped the legal domain so far? a scoping literature review. *Information*, 15(11):662.

Lucas de Castro Rodrigues Pereira, Maykon Marcos Junior, Guilherme de Brito Santos, Isabela Cristina Sabo, Thiago Raulino Dal Pont, Andressa Silveira Viana Maurmann, Luísa Bollmann, Maite Fortes Vieira, João Gabriel Mohr, Cristian Alexandre Alchini, and 1 others. 2025. Using

GPT-4o as a factor extractor for Brazilian consumer law judgments. *Artificial Intelligence and Law*, pages 1–29.

S. Shukla, R. Shankar, and S. P. Singh. 2014. Food safety regulatory model in India. *Food Control*, 37:401–413.

Marco Siino, Mariana Falco, Daniele Croce, and Paolo Rosso. 2025. Exploring LLMs applications in law: A literature review on current legal NLP approaches. *IEEE Access*.

Stavroula Skylaki, Ali Oskooei, Omar Bari, Nadja Herger, and Zac Kriegman. 2020. Named entity recognition in the legal domain using a pointer generator network. *Preprint*, arXiv:2012.09936.

Zsolt Ződi. 2024. Legal technology in the service of access to justice. *Hungarian Journal of Legal Studies*, 64(3):323–335.

A Appendix

A.1 Sample Case Output

Below is a sample output for a case titled *M/S Itc Limited vs Food Safety Inspector on 12 October, 2023*, demonstrating the structured extraction format:

<Begin Output Summary>
Petitioner: M/S Itc Limited

Respondent: FOOD SAFETY INSPECTOR, DEPARTMENT OF THE FOOD SAFETY AND STANDARDS KALABURAGI

Court: Karnataka High Court

Case overview: The case involves a writ petition filed by M/S ITC Limited against the Food Safety Inspector and the Designated Officer of the Department of Food Safety and Standards, challenging an order dated December 29, 2015, which imposed a fine for misbranding of their product, Sunfeast Yippee Noodles. The petition was heard in the High Court of Karnataka, Kalaburagi Bench.

Key facts:

- The petitioner, M/S ITC Limited, manufactures instant noodles and was accused of misbranding due to the label stating 'No Added MSG'.
- A sample was collected on June 8, 2015, and tested, with the Chief Food Analyst reporting it as misbranded.
- The petitioner claimed they were not given proper notice or the right to appeal against the Food Analyst's report.
- The Designated Officer imposed a fine of Rs. 10,000 based on the Food Analyst's report.

Legal issues and arguments:

- The petitioner argued that the product complied with all standards and that the misbranding claim was unfounded, as there was no evidence that MSG was added during manufacturing.
- They contended that the prosecution was initiated without proper inspection of the manufacturing premises and without affording them the right to appeal as per Section 46(4) of the Food Safety and Standards Act, 2006.
- The respondents maintained that the product was misbranded due to the label claim and that the prosecution was justified.

Court's reasoning:

- The court noted that the Chief Food Analyst's report indicated compliance with standards but also stated that the product was misbranded due to the label claim.
- It highlighted that there was no analytical method to determine if MSG was added or naturally present, and that prosecution should not occur without ascertaining the addition of MSG during manufacturing.
- The court found that the Designated Officer did not inspect the manufacturing unit, which was necessary to determine the validity of the misbranding claim.
- The court emphasized that the petitioner was not given the opportunity to appeal, violating principles of natural justice.

Decision or judgment: The court allowed the writ petition, quashing the order passed by the Additional District Magistrate cum Adjudicating Authority in C.C.No.316/203/54/2015-16 dated December 29, 2015, against M/S ITC Limited.

Type: Case won by Appellant
<End Output Summary>

A.2 Entity Extraction Prompt

You will be provided with a legal judgment of a food safety related case. Your goal is to extract key information following the schema provided. Please ensure that the extracted information is accurate and complete. Any references to the Food Safety and Standards Act should be extracted as "Section x of the Food Safety and Standards Act, 2006" or "Food Safety and Standards Act, 2006". Always use full forms of abbreviations, e.g. "Supreme Court" instead of "SC", or "Indian Penal Code" instead of "IPC". Names of courts should be standardized and follow correct capitalization, e.g. "Supreme Court of India", "High Court of Delhi", NOT "Supreme Court", "Delhi High Court" or "DELHI HIGH COURT".

Here is a description of the parameters to be extracted:

- court: name of the court that issued the judgment.
- petitioners: array of strings containing the names of ALL appellant(s) in the case.

- respondents: array of strings containing the names of ALL respondent(s) in the case.
- judges: array of strings containing the names of the judge(s) in the case.
- date: date of the judgment, as a string in the format "DD-MM-YYYY".
- org: array of strings containing the names of all organizations, companies, or government entities mentioned in the case, if any.
- gpe: array of strings containing the names of all geographical locations mentioned in the case, if any.
- provisions: array of strings containing the provisions of ALL statutes cited in the judgment. Provide these in the format "Section x of y". In case of references to multiple sections of the same statute, list them all separately.
- statutes: array of strings containing the names of ALL acts or laws cited in the judgment.
- precedents: array of strings containing the names of ALL precedents cited in the judgement.
- key facts: key facts about the case. This should be very concise, and include the background of the case, and the main arguments made by both parties. If no information is provided, leave this field empty.
- type of case: the type of case, e.g. bail application, civil appeal, criminal appeal, public interest litigation, etc.
- decision: the decision of the court in the case, if provided. If there is a verdict, respond with 'in favour of appellant' or 'in favour of respondent'. If not, leave this field empty.

A.3 Summary Generation Prompt

You will be provided with a legal judgment of a food safety-related case. Your goal is to provide a detailed summary of the judgment. Only include information explicitly stated in the document. Do not infer, interpret, or add new information. Use concise language while preserving all critical legal details, such as statute names, provisions, case outcomes, and involved parties. Organize the summary logically, grouping related points. For example:

- Case overview
- Key facts
- Legal issues and arguments
- Court's reasoning
- Decision or judgment

Return plain text, do not include any markdown or HTML formatting.

A.4 Cluster Labeling Prompt

<Task>
You are an expert document analyst tasked with analyzing clusters of similar documents and generating concise, descriptive labels that capture the common theme or topic of the cluster.
</Task>

<Documents>
{documents}
</Documents>

<Instructions>

1. Read through all the provided documents carefully.
2. Identify the common themes, topics, or patterns across the documents.
3. Focus on:
   - Legal subject matter (if applicable)
   - Key parties or entities involved
   - Types of cases or proceedings
   - Common legal issues or questions
   - Geographic or jurisdictional patterns
   - Temporal patterns or time periods
4. Generate a SHORT, descriptive label (3-10 words maximum) that captures the essence of the cluster.
5. The label should be:
   - Specific enough to distinguish this cluster from others
   - General enough to encompass all documents in the cluster
   - Clear and understandable to someone unfamiliar with the documents
   - Professional and concise

<Instructions>

<OutputFormat>
Return ONLY the label text without any additional explanation, prefixes, or formatting. Do NOT include phrases like "Label:", "Cluster:", or "Theme:". Just provide the descriptive label itself. Examples of good labels are "Food Safety Violations and Regulatory Actions", "Municipal Tax and Assessment Challenges", "Employment Termination and Labor Rights", "Environmental Compliance and Pollution Cases"
<OutputFormat>

A.5 Cluster Label Semantic Similarity Analysis

Tables 1, 2, and 3 provide detailed pairwise semantic similarity scores between cluster labels generated by GPT-4o, Gemini-2.5-Flash, and Qwen-3, as described in Section 6.4. Labels with SBERT similarity scores above our established threshold of 0.55 indicate semantic consensus. The high proportion of aligned labels across all model pairs validates the robustness of our clustering approach.

GPT	Gemini	SBERT Score
Appeals and Remedies Under Food Safety Act	Food Safety Act Appeals and Remedies	0.99
Bail Applications in Food Safety Violations	Food Safety Violations and Bail Applications	0.99
Food Safety Officer Recruitment Disputes	Food Safety Officer Recruitment Disputes	0.98
Food Safety Act Violations and Appeals	Food Safety and Standards Act Violations	0.93
Food Safety Compliance and Violation Cases	Food Safety and Standards Act Violations	0.91
Import and Customs Disputes	Food Safety and Import Regulation Disputes	0.90
Food Safety Violations and Legal Challenges	Food Safety and Standards Act Violations	0.90
Food Safety Violations and Legal Proceedings	Food Safety Act Cases and Quashing Proceedings	0.89
Food Safety Enforcement Challenges	Food Safety and Standards Act Violations	0.86
Food Safety Regulation Disputes	Food Safety and Import Regulation Disputes	0.86
Limitations in Food Safety Prosecutions	Food Safety Act: Limitation and Quashing	0.85
Quashing of Cases Under IPC and FSS Act	Food Safety Act Cases and Quashing Proceedings	0.83
Ethanol Production and Supply Restrictions	Ethanol Production Regulation and Sugar Supply	0.81
Trademark and Commercial Disputes	Trademark Infringement and Product Disputes	0.72
Licensing and Regulatory Challenges in Meat	Food Business Licensing Under FSS Act	0.60
Divergent / Low Consensus Matches (Score < 0.55)		
Tender and Procurement Disputes	Food Procurement and Safety Disputes	0.42
Sealing Actions and Legal Disputes	Food Safety Act Cases and Quashing Proceedings	0.37

Table 1: Semantic alignment between GPT and Gemini cluster labels.

Gemini	Qwen	SBERT Score
Food Safety and Standards Act Violations	Food Safety and Standards Act Violations	0.97
Food Safety Violations and Bail Applications	Food Safety Violations and Anticipatory Bail Cases	0.94
Food Safety Act Appeals and Remedies	Food Safety Act Enforcement and Appeals	0.93
Food Safety Act Violations and Quashment	FSS Act Violations Quashed by Precedent	0.88
Food Safety Act: Limitation and Quashing	Food Safety Act Enforcement and Appeals	0.87
Food Safety Procedural Violations and Quashing	Food Safety Prosecution Challenges	0.87
Food Safety Act Cases and Quashing	Food Safety and Standards Act Cases	0.86
Food Safety and Import Regulation Disputes	Food Import Clearance and Regulatory Compliance	0.84
Quashing Food Safety Proceedings	Food Safety Prosecution Challenges	0.81
Food Safety, Seizure, and Business Operations	Food Safety and Seizure Disputes	0.80
Food Safety Act Convictions and Sentence	FSS Act Violations Quashed by Precedent	0.78
Food Procurement and Safety Disputes	Public Procurement and Tender Disputes	0.76
Food Safety Act Cases in Madhya Pradesh	Food Safety and Standards Act Cases	0.71
Food Business Licensing Under FSS Act	Food Safety and Adulteration Cases	0.70
Food Safety Officer Recruitment Disputes	Anganwadi Workers and Officer Recruitment	0.68
Ethanol Production Regulation	Ethephon and Ethanol Regulatory Disputes	0.57
Divergent / Low Consensus Matches (Score < 0.55)		
Trademark Infringement and Product Disputes	Food Safety Licensing and Meat Business	0.38

Table 2: Semantic alignment between Gemini and Qwen cluster labels.

GPT	Qwen	SBERT Score
Food Safety Act Violations and Appeals	Food Safety and Standards Act Violations	0.97
Appeals and Remedies Under Food Safety Act	Food Safety Act Enforcement and Appeals	0.92
Bail Applications in Food Safety Violations	Food Safety Violations and Anticipatory Bail Cases	0.92
Food Safety Compliance and Violation Cases	FSS Act Violations and Regulatory Compliance	0.91
Food Safety Enforcement Challenges	FSS Act Violations and Regulatory Compliance	0.90
Food Safety Violations and Legal Challenges	Food Safety Prosecution Challenges	0.89
Licensing and Regulatory Challenges in Meat	Food Safety Licensing and Meat Business	0.88
Quashing of Cases Under IPC and FSS Act	Food Safety and Standards Act vs. IPC Cases	0.88
Food Safety Violations and Legal Proceedings	Food Safety and Standards Act Violations	0.87
Limitations in Food Safety Prosecutions	Food Safety Prosecution Challenges	0.87
Food Safety Regulation Disputes	Food Safety Act Enforcement and Appeals	0.85
Import and Customs Disputes	Food Import Clearance and Regulatory Compliance	0.84
Food Safety Officer Recruitment Disputes	Anganwadi Workers and Officer Recruitment	0.71
Ethanol Production and Supply Restrictions	Ethephon and Ethanol Regulatory Disputes	0.62
Tender and Procurement Disputes	Public Procurement and Tender Disputes	0.61
Divergent / Low Consensus Matches (Score < 0.55)		
Trademark and Commercial Disputes	Food Safety Licensing and Meat Business	0.35
Sealing Actions and Legal Disputes	Food Safety and Standards Act Litigation	0.32

Table 3: Semantic alignment between GPT and Qwen cluster labels.

Grahak-Nyay: Consumer Grievance Redressal through Large Language Models

Shrey Ganatra[1], **Swapnil Bhattacharyya**[1], **Harshvivek Kashid**[1], **Spandan Anaokar**[1], **Shruti Nair**[2], **Reshma Sekhar**[2], **Siddharth Manohar**[2], **Rahul Hemrajani**[2], **Pushpak Bhattacharyya**[1]
[1]Indian Institute of Technology Bombay
[2]National Law School of India University, Bangalore
`{ganatrashrey2002, harshvivek14, spandananao, pushpakbh}@gmail.com`
`swapnilbhyya@cse.iitb.ac.in`

Abstract

Access to consumer grievance redressal in India is often hindered by procedural complexity, legal jargon, and jurisdictional challenges. To address this, we present **Grahak-Nyay** (Justice-to-Consumers), a chatbot that streamlines the process using open-source Large Language Models (LLMs) and Retrieval-Augmented Generation (RAG). Grahak-Nyay simplifies legal complexities through a concise and up-to-date knowledge base. We introduce three novel datasets: *GeneralQA* (general consumer law), *SectoralQA* (sector-specific knowledge) and *SyntheticQA* (for RAG evaluation), along with *NyayChat*, a dataset of 303 annotated chatbot conversations. We also introduce *Judgments* data sourced from Indian Consumer Courts to aid the chatbot in decision making and to enhance user trust. We also propose **HAB** metrics (**Helpfulness, Accuracy, Brevity**) to evaluate chatbot performance. Legal domain experts validated Grahak-Nyay's effectiveness. Code and datasets are available at `https://github.com/ShreyGanatra/GrahakNyay.git`.

1 Introduction

Large Language Models (LLMs) like GPT-4 (Achiam et al., 2023) and Llama-3 (Dubey et al., 2024) have found widespread use in various domains, including finance (Zhao et al., 2024), tourism (Meyer et al., 2024), healthcare (Mishra et al., 2023), education (Lee et al., 2023), and customer support (Obadinma et al., 2022). While LLMs have been applied to legal tasks such as judgment prediction, summarization, and case retrieval (Joshi et al., 2024; Feng et al., 2024), there's a notable gap in their application to consumer law, especially for assisting individuals with everyday grievances. This is particularly crucial in India.

In India, consumer grievance redressal remains a significant challenge. Despite the efforts of the Department of Consumer Affairs[1] through initiatives like the National Consumer Helpline[2] and the e-Daakhil[3] portal, many individuals without legal expertise still struggle with filing complaints. While legal representation is not required to file a consumer complaint, the complexity of legal language, jurisdictional issues, and strict limitation periods create barriers for consumers. In the financial year 2023-2024, approximately 107,966 complaints were registered every month with the National Consumer Helpline (Ministry of Consumer Affairs), indicating a large volume of grievances that need timely attention. However, these complaints are handled by trained staff, revealing a critical gap in consumer awareness and legal literacy, and underscoring the need for tools to accelerate the grievance resolution.

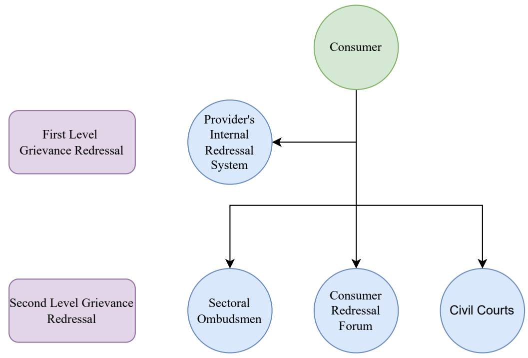


Figure 1: Two Level Grievance Redressal System in India, where the Consumer is first encouraged to approach the Provider's Internal Redressal System (like Customer Care) and then reach out to Government bodies for redressal.

Existing general-purpose chatbots like ChatGPT[4] and Claude[5] provide generic information, but lack the specific legal knowledge and interactive ca-

[1]`https://consumeraffairs.nic.in/`

[2]`https://consumerhelpline.gov.in/`
[3]`https://edaakhil.nic.in/`
[4]`https://chatgpt.com/`
[5]`https://claude.ai/`

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 53–72
December 24, 2025

pabilities needed for effective consumer grievance filing (Fig. 13). India's two-level grievance redressal system (Fig. 1) encourages direct contact with service providers, escalating to government bodies if necessary. However, complaints are frequently rejected due to issues such as as incorrect jurisdiction, misrepresentation, or failure to meet legal requirements, often stemming from a lack of understanding of consumer law (Reserve Bank of India) (Fig. 6).

To address this, we introduce **Grahak-Nyay** (Justice-to-Consumers), a chatbot designed to empower Indian consumers by providing the legal knowledge necessary to navigate the grievance redressal system. Grahak-Nyay assists in interpreting complex legal language, preparing documentation (complaint letters, forms) and guiding users through escalation procedures. Unlike general-purpose chatbots, Grahak-Nyay offers context-specific legal assistance, enabling informed action without requiring formal legal representation. By addressing the key challenges – lack of consumer law knowledge and documentation complexity – our chatbot aims to increase successful complaint filings and streamline the resolution process.

Our contributions are:

1. ***Grahak-Nyay***: A consumer grievance redressal chatbot tailored for Indian consumers, utilizing an open-source Large Language Model powered by Retrieval-Augmented Generation, aided by a concise Knowledge Base with the latest information (Section 3).

2. ***GeneralQA***: A question-answer dataset based on general consumer laws; ***SectoralQA***: A question-answer dataset based on sector-wise knowledge of consumer laws; and ***SyntheticQA***: A question-answer dataset to evaluate the performance of RAG (Section 2.1).

3. ***Judgments Data***: An annotated corpus of 570 Indian Consumer Court judgments along with summaries and categories, used to enhance user trust (Section 2.2).

4. ***NyayChat***: A dataset containing 300 annotated conversations between users and the chatbot based on various issues and complaints. Each conversation averages 32 turns and 3,475 tokens, demonstrating the depth and richness of the interactions. (Section 2.3).

Statistic	**Value**
Total Conversations	303
Average Turns per Conversation	32.01
Average Tokens per Conversation	3475.26

Table 1: Statistics of the NyayChat dataset, which consists of simulated conversations addressing consumer law grievances.

5. **HAB** metrics, to assess the quality of chatbot conversations based on **H**elpfulness, **A**ccuracy, and **B**revity. We conduct human-based evaluation and experiments on reference-free automatic evaluation of conversations using various *LLM-based* evaluators based on the HAB metrics (Section 4).

2 Dataset

We introduce four datasets: three forming the knowledge base for Retrieval-Augmented Generation (RAG), and one containing simulated user-chatbot conversations. For more detail on the dataset curation process, refer to Section A.3.

2.1 Knowledge Base

2.1.1 GeneralQA

This dataset provides a structured overview of the Consumer Protection Act 2019 (Ministry of Law and Justice), consisting of 53 question-answer pairs. It covers the Act's key provisions and defines fundamental concepts like 'consumer' and 'goods,' clarifying the chatbot's operational scope. This dataset establishes the foundational legal framework for the chatbot.

2.1.2 SectoralQA

This dataset encompasses detailed information from 23 sectors, including real estate, telecommunications, sports and recreation. The corpus comprises 889 questions and answers, each tailored to address sector-specific concerns. A breakdown of the sector-wise distribution, along with the general corpus, is provided in Fig. 7.

2.1.3 SyntheticQA

To assess our RAG system, we generated five user questions that yield the same answer across both the general corpus and the sector-specific corpus, using the Llama-3.1-70B-Instruct[6] model. This

[6] https://huggingface.co/meta-llama/Llama-3.1-70B-Instruct

User: Builder is not giving possession, any past judgments that can help me, please cite them?

Grahak-Nyay: I'd be happy to help you with that. In a case where possession was offered after a long delay, but the complainants refused to take possession as they no more needed the flat after such delay, it was held that only the earnest money/initial deposit paid by the buyer can be forfeited by the builder in case the buyer does not want to continue in the project due to any default of the builder. **(CC/438/2019/2020, NCDRC)**
Additionally, in a case where the builder delayed handing over flats after collecting a significant amount of money, the National Consumer Disputes Redressal Commission (NCDRC) held that the builder is liable for deficiency in service and must compensate the buyers for the delay. **(FA/347 of 2021/2024, NCDRC)**
In another case, where possession was delayed for nearly 7 years, the NCDRC held that the delay in possession was a deficiency of service, justifying compensation. **(CC/197/2013/2022 NCDRC)**
These judgments may be relevant to your case, but please note that each case is unique, and the outcome may vary depending on the specific circumstances.
Would you like to know more about the process of filing a complaint against the builder or drafting a legal notice?

Figure 2: Chat example from Grahak Nyay using the Judgments Database to answer user queries

amounts to generating 4734 (in some cases, 6 instead of 5 questions) questions. The answers were generated using the Llama-3.1-8B-Instruct[7] model, keeping the corresponding answer in the context. We call this dataset as **SyntheticQA**. The prompt used for generation can be found in Fig.12.

2.2 Judgments

We introduce a novel, expert-annotated corpus of judgments from Consumer Courts in India, a critical resource for advancing NLP research in the legal domain. This corpus comprises 570 judgments, meticulously curated and spanning 23 distinct sectors of Indian consumer law. Each judgment has been summarized and categorized by legal experts, ensuring high-quality annotations and domain-specific relevance.

To the best of our knowledge, this represents the first publicly available dataset of its kind, addressing a significant gap in resources for legal NLP, particularly within the Indian judicial context. The dataset's immediate utility is demonstrated in its application within our legal chatbot (Figure 2), where it enables informed decision-making and allows for the citation of pertinent case law to enhance user trust and transparency.

Beyond this specific application, the corpus holds substantial potential for a range of NLP tasks. Its rich annotations and structured nature make it highly suitable for benchmarking and developing models for legal text summarization, case classification, legal information retrieval, and potentially for predicting case outcomes or identifying relevant legal precedents. We believe this dataset will be an invaluable asset to the NLP community, fostering further research and development in the underexplored intersection of artificial intelligence and consumer law.

[7] https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct

2.3 NyayChat

This dataset includes 303 simulated conversations meticulously crafted by a team of legal experts specializing in various sectors such as e-commerce, medical negligence, railways, airlines, and more. Each conversation mirrors a real-world interaction between a user and the chatbot, addressing specific grievances that fall under the purview of consumer law. Each conversation averages 32.01 turns and 3,475.26 tokens (Tab. 1), demonstrating the depth and richness of the interactions. This dataset serves as a valuable benchmark for advancing research in conversational AI, particularly in the domain of user grievance redressal.

3 Methodology

3.1 Retrieval-Augmented Generation

Retrieval Augmented Generation (RAG) is a prominent approach used in real-world applications for grounding large language model (LLM) generations in up-to-date and domain-specific knowledge. It has been observed (Lazaridou et al., 2022; Shuster et al., 2021; Ren et al., 2023) that RAG reduces hallucinations and improves answer quality, without the need for highly expensive and sometimes fragile domain-specific fine-tuning.

A typical RAG framework (Fig. 3) involves a retrieval system that fetches documents that are relevant to the query. These documents are then used as context, prompting the LLM (Fig. 4) to generate the required response. For our chatbot, we

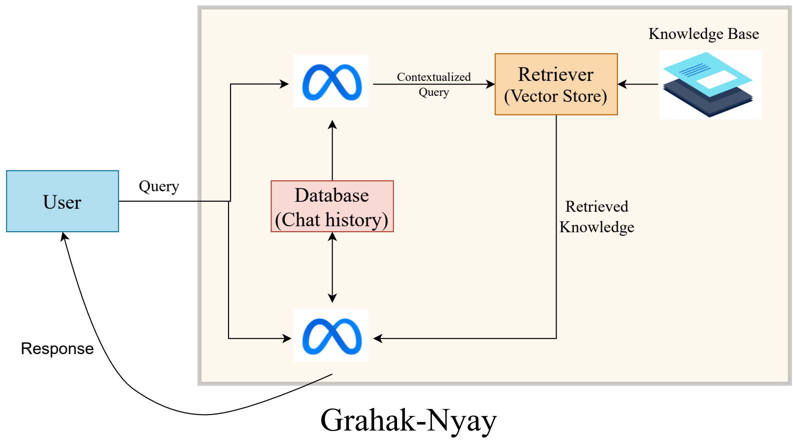


Figure 3: Architecture of *Grahak-Nyay* chatbot. The user query is first contextualised using an LLM to form an independent query to perform RAG. The retrieved knowledge is then given to the LLM along with the chat history to generate an appropriate response to the user query.

also include the chat history as part of the prompt. The RAG framework is divided into the following parts in our architecture.

3.1.1 Chunking

We observed that fixed-length chunking proved inadequate for our purposes. Long answers were often split across multiple chunks, resulting in the loss of crucial information during retrieval. Conversely, grouping multiple short answers in a single chunk introduced unnecessary noise. As a result, we adopted a chunking strategy where each chunk contains only one Question-Answer pair, ensuring clarity and precision in information retrieval.

3.1.2 Query Rewriting

To implement RAG, it is essential for each query to be properly contextualized. To accomplish this, we utilized Llama-3.1-8B-Instruct along with the instruction provided in Fig.9. Initially, we observed that the LLM answered the query directly rather than rewriting it as intended. To improve performance, we utilized one-shot prompting by incorporating a single example in the prompt, which led to significantly better results.

3.1.3 Retriever

The chunks are embedded using mixedbread-ai/mxbai-embed-large-v1[8] and stored in a vector-store. The retriever is responsible for extracting relevant chunks using the query. We use cosine similarity to extract the best four chunks and use it for response generation.

[8] https://huggingface.co/mixedbread-ai/mxbai-embed-large-v1

3.1.4 Generation

The chatbot is meant to be interactive and conversational. Hence, for each user input, while keeping the retrieved documents as a reference, the chatbot also takes into account the chat history. This ensures that all responses are relevant and grounded in the contextual history of the issue. We use Llama-3.1-8B-Instruct model to generate our responses.

3.2 Hallucination

A significant barrier to the wide use of LLMs in multiple domains is their tendency to hallucinate. It has been observed that in spite of clear instructions, LLM generates text which might be false or irrelevant. RAG-based approaches help to tremendously reduce this phenomenon, but even then, hallucination remains a major concern.

A domain-specific chatbot like our Grahak-Nyay chatbot implies that RAG is responsible for providing much of the context and domain knowledge that will be utilized for the chat. In such cases, the LLM must generate content based on the RAG Corpus only and never contradict it. For this purpose, we modified the prompt where we specified the LLM to answer any out-of-corpus question by stating that it does not know the answer.

4 Evaluation

We strongly believe that any user-facing chatbot should help the user address the query, be accurate while doing so, and keep the user engaged. We assess the quality of chatbot conversations using **HAB metrics**: **H**elpfulness, **A**ccuracy, and **B**revity.

```
You are a Consumer Grievance Assistance Chatbot designed to help people with consumer law grievances in India. Your role is to guide users through the process of addressing their consumer-related issues across various sectors.

Core Functionality:

Assist with consumer grievances ...

Conversation Flow:

1.Greet the user and ask about their consumer grievance.

2. If ...

...

Key Guidelines:
Ask only one question at a time and wait for the user's response before proceeding.
...

Use only the facts/names provided in the context or by the user.

Don't let the user know you answered the question using the context.

\n\n

Here is the Context:

{context}
```

Figure 4: Part of system prompt designed for Grahak-Nyay Chatbot with a structured conversation flow: it gathers grievance details step-by-step, offers remedies under Indian consumer law, assists in drafting legal documents (e.g., notices, complaints), guides users on using the National Consumer Helpline and e-daakhil portal, and provides tailored responses while strictly limiting interactions to consumer-related issues. For entire prompt see Figure 14
.

Reference-based evaluation						Reference-free evaluation		
ROUGE-1	**ROUGE-2**	**ROUGE-L**	**BERTScore**	**METEOR**	**BLEU**	**Helpfulness**	**Accuracy**	**Brevity**
66.9	41.1	33.2	90.9	41.9	37.4	4.65	3.61	3.12

Table 2: Performance of *Grahak-Nyay chatbot* on Reference-based and Reference-free evaluation. We evaluated the Grahak-Nyay chatbot on 65 conversations for which reference was available. We performed LLM-based automatic evaluation on HAB metrics on the 5-point Likert scale using the gpt-4o-mini model.

Dataset	BLEU	ROUGE-1	ROUGE-L	BERTScore	Ans-Rel.	Faithfulness
SectoralQA	49.38	64.20	60.39	90.94	7.44	8.58
GeneralQA	49.45	66.66	63.74	95.18	7.35	9.02
SyntheticQA	31.04	48.37	40.44	87.93	7.48	9.30

Table 3: Performance on BLEU, ROUGE, and BERTScore, along with automatic evaluation using RAGAS assessment based on Answer Relevance (Ans-Rel.) and Faithfulness metrics across three datasets.

HAB metrics allow us to assess how effectively the chatbot addresses user issues and provides accurate information and how concisely it communicates these responses. We also qualitatively assess the chatbot performance (Fig. 18, 19, 20 and 21) where multiturn conversations of human followed by chatbot has been presented.
The HAB metrics are defined as follows:

- **Helpfulness**: This metric assesses how helpful the chatbot was in resolving the user's issue or query. It evaluates the chatbot's ability to understand the user's problem and provide actionable, relevant, and clear resolution.

- **Accuracy**: This metric evaluates the correctness of the information provided by the chatbot in response to user queries, ensuring that the responses are factually accurate and reliable.

- **Brevity**: This metric measures the conciseness of the chatbot's responses, ensuring efficient communication without unnecessary elaboration. It ensures efficient communication by focusing on delivering essential information straight to the point while avoiding excessive questioning or verbosity.

4.1 Human Evaluation of other chatbots

Using the HAB metric, we evaluated publicly available chatbots, including `ChatGPT-4.0`, `Claude-3.5`, `Llama-3.1-405b-128k`, and `Llama-3.1-8b-128k`, with assessments conducted by human *legal* experts on 5-point Likert scale. The analysis revealed that the *Grahak-Nyay* chatbot surpassed all other chatbots on the HAB metrics (Fig. 5).

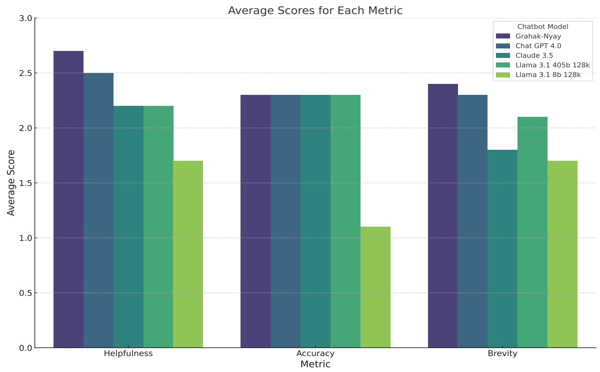


Figure 5: Benchmarking of *Grahak-Nyay* against other chatbots on HAB metrics by human legal experts. Grahak-Nyay outperforms in Helpfulness and Brevity. It performs similar to other larger models while outperforming the base model in terms of Accuracy.

4.2 Evaluation of Grahak-Nyay chatbot

We present the detailed results from the evaluation of 65 chats obtained using the Grahak-Nyay chatbot in Table 2, categorized into two groups: Reference-based and Reference-free evaluations. For these 65 chats, reference responses annotated by the legal experts were available, enabling the application of Reference-based metrics. Additionally, for the Reference-free evaluation, we utilized HAB metrics to assess the chatbot's performance in providing relevant and concise responses. We used the best performing model, gpt-4o-mini model, which demonstrated the highest correlation with human evaluations, for the LLM-based assessment of the HAB metrics on 5-point Likert scale (Appendix A.1).

4.3 Evaluation of Retrieval-Augmented Generation

We evaluate our Retrieval-Augmented Generation (RAG) system by asking questions from GeneralQA, SectoralQA, and SyntheticQA. The system is assessed using BLEU, ROUGE, and BERTScore, along with automatic evaluation metrics such as Faithfulness and Answer Relevance using **RAGAS** (Es et al., 2023). Detailed results are presented in Table 3. Faithfulness (Fig. 10) measures whether the generated answer is grounded in the provided context. Answer Relevance (Fig. 11) evaluates how well the generated answer addresses the given question. To assess relevance, we compare the generated response with the ground truth. We use the gpt-4o-mini model as an evaluator. Faithfulness and Answer Relevance are scored on a 0-10 scale, while other metrics are measured on a 0-100 scale.

5 Deployment

We utilize the Text Generation Inference (TGI)[9] toolkit (v3.2.1) to serve the `Llama-3.1-8B-Instruct` model. TGI provides a production-ready server with features crucial for real-world deployment, including continuous batching of incoming requests for increased throughput, prefix caching to reduce redundant computations, and token streaming using Server-Sent Events (SSE) for a responsive user experience.

Our current deployment utilizes a single NVIDIA A100 GPU with 40GB of memory through its official docker image (Fig. 8).

We plan on incorporating auto-scaling and adding high availability to handle potential outages.

6 Conclusions

In this work, we introduced our *Grahak-Nyay* chatbot to address consumer grievances in various sectors. We evaluated the chatbot performance using traditional NLP metrics, automated evaluation by LLMs, and human evaluation by legal experts. Using a RAG-based framework and prompts designed to prevent hallucinations, the chatbot demonstrated the ability to handle consumer grievances in an approachable and informative way. The chatbot presents an opportunity for many people who are hesitant to take action on their consumer complaints due to a lack of complete information and help them get justice.

7 Limitations

A primary concern is the inherent tendency of large language models to generate hallucinated or inaccurate information, particularly when dealing with specific legal provisions, case precedents, or procedural requirements. The model may confidently present incorrect statutory references, fabricate non-existent legal remedies, or provide outdated guidance that no longer aligns with current consumer protection laws. Furthermore, the system's knowledge base requires continuous updates to reflect amendments in consumer protection legis-

[9] `https://huggingface.co/docs/text-generation-inference/en/index`

lation, new regulatory guidelines, evolving judicial interpretations, and changes in forum procedures.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Shahul Es, Jithin James, Luis Espinosa-Anke, and Steven Schockaert. 2023. Ragas: Automated evaluation of retrieval augmented generation. *Preprint*, arXiv:2309.15217.

Yi Feng, Chuanyi Li, and Vincent Ng. 2024. Legal case retrieval: A survey of the state of the art. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6472–6485, Bangkok, Thailand. Association for Computational Linguistics.

Abhinav Joshi, Shounak Paul, Akshat Sharma, Pawan Goyal, Saptarshi Ghosh, and Ashutosh Modi. 2024. IL-TUR: Benchmark for Indian legal text understanding and reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11460–11499, Bangkok, Thailand. Association for Computational Linguistics.

Angeliki Lazaridou, Elena Gribovskaya, Wojciech Stokowiec, and Nikolai Grigorev. 2022. Internet-augmented language models through few-shot prompting for open-domain question answering. *Preprint*, arXiv:2203.05115.

Seungjun Lee, Yoonna Jang, Chanjun Park, Jungseob Lee, Jaehyung Seo, Hyeonseok Moon, Sugyeong Eo, Seounghoon Lee, Bernardo Yahya, and Heuiseok Lim. 2023. PEEP-talk: A situational dialogue-based chatbot for English education. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 190–207, Toronto, Canada. Association for Computational Linguistics.

Sonia Meyer, Shreya Singh, Bertha Tam, Christopher Ton, and Angel Ren. 2024. A comparison of llm finetuning methods & evaluation metrics with travel chatbot use case. *Preprint*, arXiv:2408.03562.

Ministry of Consumer Affairs. 2024. Nearly 1,07,966 average number of dockets registered on monthly basis with national consumer helpline between april 2024-june 2024.

Ministry of Law and Justice. Consumer protection act, 2019.

Kshitij Mishra, Priyanshu Priya, and Asif Ekbal. 2023. PAL to lend a helping hand: Towards building an emotion adaptive polite and empathetic counseling conversational agent. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12254–12271, Toronto, Canada. Association for Computational Linguistics.

Stephen Obadinma, Faiza Khan Khattak, Shirley Wang, Tania Sidhorn, Elaine Lau, Sean Robertson, Jingcheng Niu, Winnie Au, Alif Munim, and Karthik Raja Kalaiselvi Bhaskar. 2022. Bringing the state-of-the-art to customers: A neural agent assistant framework for customer service support. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 440–450, Abu Dhabi, UAE. Association for Computational Linguistics.

Yubing Ren, Yanan Cao, Ping Guo, Fang Fang, Wei Ma, and Zheng Lin. 2023. Retrieve-and-sample: Document-level event argument extraction via hybrid retrieval augmentation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 293–306, Toronto, Canada. Association for Computational Linguistics.

Reserve Bank of India. 2019. Annual report of the banking ombudsman scheme and ombudsman scheme for non-banking financial companies for the year 2018-19.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. *Preprint*, arXiv:2104.07567.

Yiyun Zhao, Prateek Singh, Hanoz Bhathena, Bernardo Ramos, Aviral Joshi, Swaroop Gadiyaram, and Saket Sharma. 2024. Optimizing llm based retrieval augmented generation pipelines in the financial domain. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 6: Industry Track)*, pages 279–294.

A Appendix

A.1 Automated Evaluation of Grahak-Nyay chatbot using LLMs

To reduce human effort in evaluating the chatbot according to HAB metrics, we employ LLM-based automatic evaluation. The LLM evaluators are instructed to assign scores on a 5-point Likert scale and provide detailed explanations for their assigned scores using the structured prompt (Fig. 15, 16, and 17). The prompt includes task description, scoring instructions based on the HAB metrics, as well as the conversation which is to be evaluated and the context[10].

We evaluated 75 conversations for which we have human-evaluated data available in binary form (Yes, if the metric is followed, No if not), on the HAB metrics, using different LLMs sourced from HuggingFace[11] and Groq[12]. The table 4 summarizes the performance of LLM-based evaluators for HAB metrics. We applied point biserial correlation to assess the relationship between the available binary human evaluation and the ordinal LLM scores from the 5-point Likert scale. This correlation is particularly useful in determining how well the LLM evaluations align with the binary outcomes. Additionally, we used Spearman correlation to evaluate the rank order of scores, providing further insights into the agreement between human and LLM evaluations. The Llama-3.1-70B model outperformed other open-source models across all three metrics, and gpt-4o-mini achieved the highest point biserial correlation and Spearman's correlation coefficients with *p-value*$<$*0.05*, indicating its superior effectiveness.

A.2 Human evaluation of Grahak-Nyay chatbot

To assess the performance of our chatbot and benchmark it against several other systems, we conducted a human evaluation of the chatbot dialogues based on the HAB metrics as outlined in 5. This evaluation was performed by a group of legal experts from the XX who were provided with a predefined set of evaluation criteria.

The evaluation of each conversation was conducted using the following rubric:

[10] Context is passed only for the *Accuracy* metric.
[11] https://huggingface.co
[12] https://groq.com/

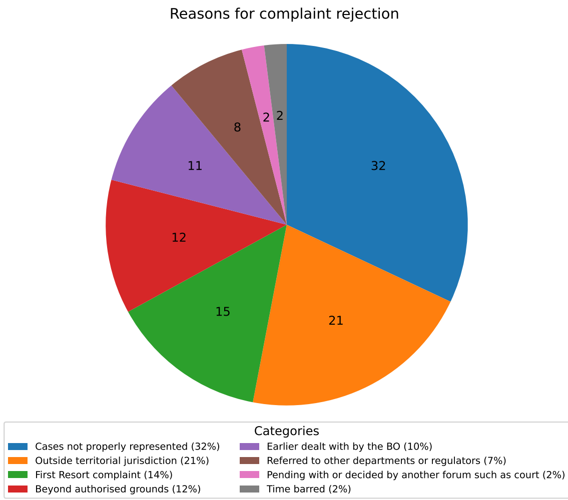


Figure 6: Reasons for Banking Ombudsman rejecting complaints in FY 2018-19. Cases not properly represented are the major reason for the rejection of complaints, followed by outside proper jurisdiction.

A.2.1 Helpfulness

Statement: The chatbot delivers meaningful assistance that contributes to resolving the user's issue.

- **Score 5** - Strongly Agree: The chatbot fully addressed the issue or provided explicit, actionable steps for resolution.
- **Score 4** - Agree: The chatbot resolved the issue to a large extent, though minor additional guidance was required.
- **Score 3** - Neutral: The chatbot provided some assistance, but the response was insufficient to fully resolve the issue.
- **Score 2** - Disagree: The chatbot's assistance was incomplete and omitted key information.
- **Score 1** - Strongly Disagree: The chatbot's response was irrelevant or ineffective in resolving the issue.

A.2.2 Accuracy

Statement: The chatbot provides precise and reliable information, including correct references such as websites, phone numbers, and legal details.

- **Score 5** - Strongly Agree: All information provided is entirely accurate and contextually appropriate.
- **Score 4** - Agree: Most information provided is accurate, with only minor, non-critical inaccuracies.

Models	Helpfulness		Accuracy		Brevity	
	r_{pb}	ρ	r_{pb}	ρ	r_{pb}	ρ
Gemma-2-9B	0.256	0.242	0.113	0.102	0.183	0.182
Llama-3.1-8B	0.386	0.246	0.225	0.213	0.154	0.153
Mixtral-8x7B	0.557	0.490	0.205	0.207	0.159	0.141
Llama-3.1-70B	<u>0.689</u>	<u>0.627</u>	**0.461**	<u>0.430</u>	<u>0.430</u>	<u>0.418</u>
gpt-4o-mini	**0.719**	**0.687**	<u>0.459</u>	**0.465**	**0.473**	**0.435**

Table 4: Performance metrics for various models based on Helpfulness, Accuracy, and Brevity metrics. Each metric includes point biserial correlation (r_{pb}) and Spearman's rank correlation coefficient (ρ) scores for each model. The best scores are bolded, and the second-best scores are underlined.

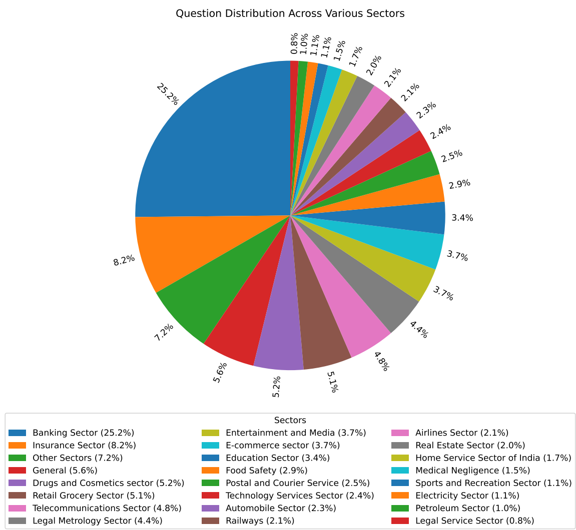


Figure 7: Distribution of corpus questions across different consumer sectors

- **Score 3**: Neutral - The chatbot provided accurate information, but there were notable factual errors.
- **Score 2**: Disagree - The response contained multiple inaccuracies that could mislead the user.
- **Score 1**: Strongly Disagree - The information provided was largely or completely incorrect and misleading.

A.2.3 Brevity

Statement: The chatbot communicates efficiently, offering clear and concise responses without superfluous information or unnecessary queries.

- **Score 5** - Strongly Agree: The response was succinct and included only the essential information.
- **Score 4** - Agree: The response was mostly concise, with minor extraneous details.
- **Score 3** - Neutral: The response included some unnecessary details or questions, reducing conciseness.
- **Score 2** - Disagree: The response was overloaded with irrelevant or redundant information, causing potential confusion.
- **Score 1** - Strongly Disagree: The response was excessively long and contained irrelevant or unnecessary information.

A.2.4 Evaluation Procedure

The evaluation of the Grahak-Nyay chatbot was conducted by two independent experts, using a blind evaluation methodology. The experts assessed 65 chats that were utilized in the automatic evaluation phase and assigned scores to each chat in terms of the HAB metrics. The evaluation of the remaining four chatbots, namely `ChatGPT-4.0`, `Claude-3.5`, `Llama-3.1-405b-128k`, and `Llama-3.1-8b-128k`, was conducted by three legal experts, following a similar blind methodology, using a representative subset of dialogues from each chatbot.

A.3 Dataset Details and Curation Process

A.3.1 GeneralQA and SectoralQA

The Knowledge Base consists of two core parts. The first is a GeneralQA on Consumer Protection in India, synthesizing general consumer grievance information into 52 question-answer pairs. These Q&A pairs span a wide range of consumer protection topics, including the Consumer Protection Act of 2019, definitions of a "consumer", details on filing consumer complaints, and overviews of online and offline redressal avenues. It also contains contact information for National, State, and District Consumer Dispute Redressal Commissions. By filtering out overly technical content, the focus remains on practical guidance: what counts as a consumer grievance, how to initiate legal action, and strategic advice on whether to send a notice or file directly in a consumer forum.

Alongside the GeneralQA, the SectoralQA includes 27 sector-specific documents, each addressing a specific consumer-related domain such as Banking, Telecom, or Insurance. Each of these contains around 30 question-answer pairs, with the total corpus having 835 Q&A pairs.

A.3.2 NyayChat

In addition to the Knowledge Base, NyayChat was developed to support detailed examples and evaluations. This dataset provides curated, real-world scenarios and user queries, allowing the LLM to be rigorously trained and tested on realistic problem statements. The dataset consists of synthetic conversations that simulate ideal interactions between the Grahak-Nyay and users seeking remedies under consumer protection laws. These synthetic chats were drafted by law students and underwent review by two legal experts. The starting point for each chat was a real-world case sourced from a database of 1,200 District Consumer Disputes Redressal Commission (DCDRC) judgments in India. A random sample of these judgments was selected and students were instructed to reimagine each situation as if they just encountered their grievances and were looking for an immediate resolution. This approach helped capture authentic and context-rich interactions that reflect real consumer disputes.

A.3.3 Sources for building the Corpus

The corpus' primary sources include official government websites (such as the Department of Consumer Affairs), regulatory authority portals, and publicly accessible laws, guidelines, and circulars related to consumer rights. It also draws information from published commentaries, Frequently Asked Questions (FAQs), and guides prepared by legal experts in the field of consumer law. By distilling over 1,500 pages of these materials into around 52 question-answer pairs, it prioritizes relevant topics for everyday consumer grievances.

A.3.4 Research teams and review process

The Corpus is collected and curated by the research team at the XX, trained in empirical and doctrinal research on legal and policy instruments and systems in India, alongside specific expertise in technology law. The project team collaborated closely with the Chair for Consumer Law and Practice at XX to determine the most effective way to compile and shape the textual corpus that would act as the knowledge base. Feedback from these experts helped refine the content so it would enhance the Large Language Model's (LLM) performance.

```
model=meta-llama/Llama-3.1-8B-Instruct
volume=$PWD/data # share a volume with the Docker container

docker run --gpus all --shm-size 40g -p 8080:80 -v $volume:/data \
    ghcr.io/huggingface/text-generation-inference:3.2.1 \
    --model-id $model
```

Figure 8: Script to run TGI server on Nvidia-GPU using official docker image

Given a chat history and the latest user question which might reference context in the chat history, formulate a standalone question which can be understood without the chat history. Do NOT answer the question, just reformulate it if needed and otherwise return it as is.
For example:
Chat History:
Human: What is Task Decompostion?
AI: Task Decomposition is the process of breaking down a complex task into smaller and simpler steps. This is achieved through a technique called Chain of Thought (CoT), which instructs the model to ẗhink step by stepänd utilize more test-time computation to transform big tasks into multiple manageable tasks.
Question: What are some of the ways of doing it?
Contextualized Question: What are some of the ways of doing Task Decompositon?

Figure 9: One-Shot Prompt for Query Rewriting to contextualise the query to perform RAG

```
Faithfulness measures the information consistency of the answer against the
given context. Any claims that are made in the answer that cannot be deduced
from context should be penalized. Given an answer and context, assign a score
for faithfulness in the range 0-10.
Format of output is:- "Faithfulness = Score out of 10"
No other output should be produced
context: [context]
answer: [answer]
```

Figure 10: Prompt for Automatic Evaluation (Faithfulness) of RAG

```
Answer Relevancy measures the degree to which a response directly addresses
and is appropriate for a given question.
It penalizes the presence of redundant information or incomplete answers given a
question. Given a question and two answers, mark a score in the range of 0-10
for answer 2 depending on how completely it answers the question
and is relevant with respect to answer 1
Format of output is:- "Relevance = Score out of 10"
question: [question]
answer 1: [answer 1]
answer 2: [answer 2]
```

Figure 11: Prompt for Automatic Evaluation (Answer Relevance) of RAG

```
You are given a frequently asked question (FAQ) and the answer to the question
in the consumer protection context.
Your task is to generate 5 questions that are likely to be asked by a consumer,
given the FAQ and the answer. Give each question in a new line.
------------------------------------------------
You are given the following example FAQ and answer:
Question: When was the Consumer Protection Act 2019 enacted?
Is the Consumer Protection Act 1986 still valid?
Answer: The Consumer Protection Act 2019 was notified on August 9, 2019.
However, it came into effect on July 20, 2020. This Act replaced the Consumer
Protection Act 1986 to address the contemporary issues faced by consumers and
to streamline the consumer grievance redressal process in India.
The Consumer Protection Act 1986 was the first significant legislation in India
aimed at protecting consumer interests. It provided for the establishment of
consumer councils and other authorities for the settlement of consumers'
disputes and for matters connected therewith. It has now been repealed by the
Consumer Protection Act 2019. Under Section 107(2) of the Consumer Protection
Act 2019, all disputes arising out of the previous act shall now be governed
by the new act.
Generated Questions:
What are the key differences between the Consumer Protection Act 1986 and the
Consumer Protection Act 2019?
Why was the Consumer Protection Act 1986 replaced by the Consumer Protection
Act 2019?
Are there any provisions in the Consumer Protection Act 2019 that address
modern consumer issues?
If a dispute arose under the Consumer Protection Act 1986, will it now be
handled under the 2019 Act?
What improvements were made in the grievance redressal process in the Consumer
Protection Act 2019 compared to the 1986 Act?
------------------------------------------------
```

Figure 12: One-Shot prompt to generate SyntheticQA dataset

```
User: Hi, I have an issue with my insurance policy. How do I resolve it?
On Monday, in the morning hours, I loaded the paddy bags and dry paddy grass
onto my vehicle and was traveling back to my village Gudivada. On the way
accidentally the electric wires touched the load on the said vehicle and
caught fire. Immediately I reported the matter to the Station Fire Officer,
A.P. State Disaster Response and Service Department, Fire Station,
Vizianagaram, but by then the said load and vehicle got completely burnt
during the accident. Subsequently I reported the incident in the police
station and got the FIR registered. When I claimed for the insurance the
Insurance company refused it.

Assistant: I'm sorry to hear about the issue with your insurance claim. To
resolve this situation, here's a step-by-step guide on how to proceed:
1. Check the coverage terms: Thoroughly review your insurance policy to
understand what is covered, especially in relation to vehicle damage caused
by fire. Verify if there are any exclusions or clauses that the insurance
company might be relying on to deny your claim.
Look for specific exclusions: Certain events (like electrical fires or
carrying certain types of loads) may be excluded, and it's important to see
if the denial falls under one of these exclusions.
2. Request a Written Explanation from the Insurance Company:
Formal denial: Ask the insurance company to provide a formal letter explaining
why your claim was denied.
Details on policy violation: Ensure the letter outlines any specific policy
clauses that led to the denial. You will need these details to challenge the
denial.
3. Gather Documentation:
 ... more information ...
4. Reapply or Appeal the Claim:
... more information ...
5. Contact the Insurance Ombudsman:
... more information ...
6. Legal Action (If Necessary):
... more information ...
7. Communicate with the Insurer:
... more information ...
By following these steps, you should be able to either resolve the issue
directly with the insurer or take the matter to a regulatory authority for
further assistance. Would you like help drafting an appeal letter or
reviewing specific clauses in your policy?
```

Figure 13: A snapshot of chat between a user and ChatGPT

```
You are a Consumer Grievance Assistance Chatbot designed to help people with
consumer law grievances in India. Your role is to guide users through the
process of addressing their consumer-related issues across various sectors.
Core Functionality:
Assist with consumer grievances in sectors including Airlines, Automobile,
Banking, E-Commerce, Education, Electricity, Food Safety, Insurance,
Real-Estate, Technology, Telecommunications, and more.
Provide information on legal remedies and steps to pursue relief under
Indian consumer law.
Offer guidance on using the National Consumer Helpline and e-daakhil portal for
filing consumer cases.
Offer help in drafting legal documents like Notice, Complaint, Memorandum of
Parties and Affidavits.
Conversation Flow:
1.Greet the user and ask about their consumer grievance.
2.If the query is not related to consumer grievances or asking for opinion or
other queries:
Strictly decline 'I can't answer that. I can help you with consumer-related
issues.' and ask for a consumer grievance-related query. Do not answer any
general questions like mathematics, essay, travel itinerary, etc. Do not give
opinions. Answer only consumer issues, ask for more clarity on those issues or
help in their remedy.
3.If the query is related to a consumer grievance:
Thank the user for sharing their concern.
Ask one question at a time to gather more information:
a. Request details about what led to the issue (if cause is not clear).
b. Ask the user for the time of incident. Statue of limitations is 2 years. If
the incident is more than 2 years old warn the user regarding the same. Today's
date is {date}
c. Ask for information about the opposing party (if needed).
d. Inquire about desired relief (if not specified).
4.Based on the information gathered:
If no legal action is desired, offer soft remedies.
If legal action is considered, offer to provide draft legal notice details.
5.Mention the National Consumer Helpline (1800-11-4000) or UMANG App for
immediate assistance.
6.Offer to provide a location-based helpline number if needed.
7.Ask if there's anything else the user needs help with.
Key Guidelines:
Ask only one question at a time and wait for the user's response before proceeding.
Tailor your responses based on the information provided by the user.
Provide concise, relevant information at each step.
Always be polite and professional in your interactions.
Use only the following pieces of retrieved context to answer the question if giving out
information.
If user asks any question which requires information like address, contact details or details of
organisation, give information only if it is present in the context
If user asks for any information like address, contact details or details of organisation that is
not in context, tell that you do not have this information and suggest ways he can obtain this
information.
Use only the facts/names provided in the context or by the user.
Don't let the user know you answered the question using the context.
\n\n
Here is the Context:
{context}
```

Figure 14: System Prompt guiding the flow of our chatbot. Core Functionality entails the task of the chatbot, Conversation Flow describes the style for conversation with the user to be more helpful while Key Guidelines contains instruction to adhere to the context provided to mitigate hallucination.

```
Task Description: You will evaluate a conversation between a user and a Consumer
Grievance Chatbot. Your task is to assess how helpful the chatbot was in
assisting the user with their issue or query. Helpfulness refers to the
chatbot's ability to understand the user's problem and provide an actionable,
relevant, and clear resolution or guidance.

Evaluation Criteria:
The task is to judge the extent to which the metric is followed by the
conversation.
Following are the scores and the evaluation criteria according to which scores
must be assigned.
<score>1</score> - The chatbot's response was irrelevant or not helpful at all in
resolving the issue.
<score>2</score> - The chatbot provided only partial assistance and left out
important details.
<score>3</score> - The chatbot gave some helpful information, but it was not
enough to resolve the issue entirely.
<score>4</score> - The chatbot mostly resolved the issue, but some minor
additional guidance was needed.
<score>5</score> - The chatbot fully resolved the issue or provided clear steps
for resolution.

Instructions: Please assign a score strictly based on the evaluation criteria.
Provide a detailed explanation justifying the score. The score must be
presented within <score></score> tags only.

Example of response format:
1. Detailed explanation of evaluation.
2. Final score: Score- <score>[1-5]</score>

{conversation}
```

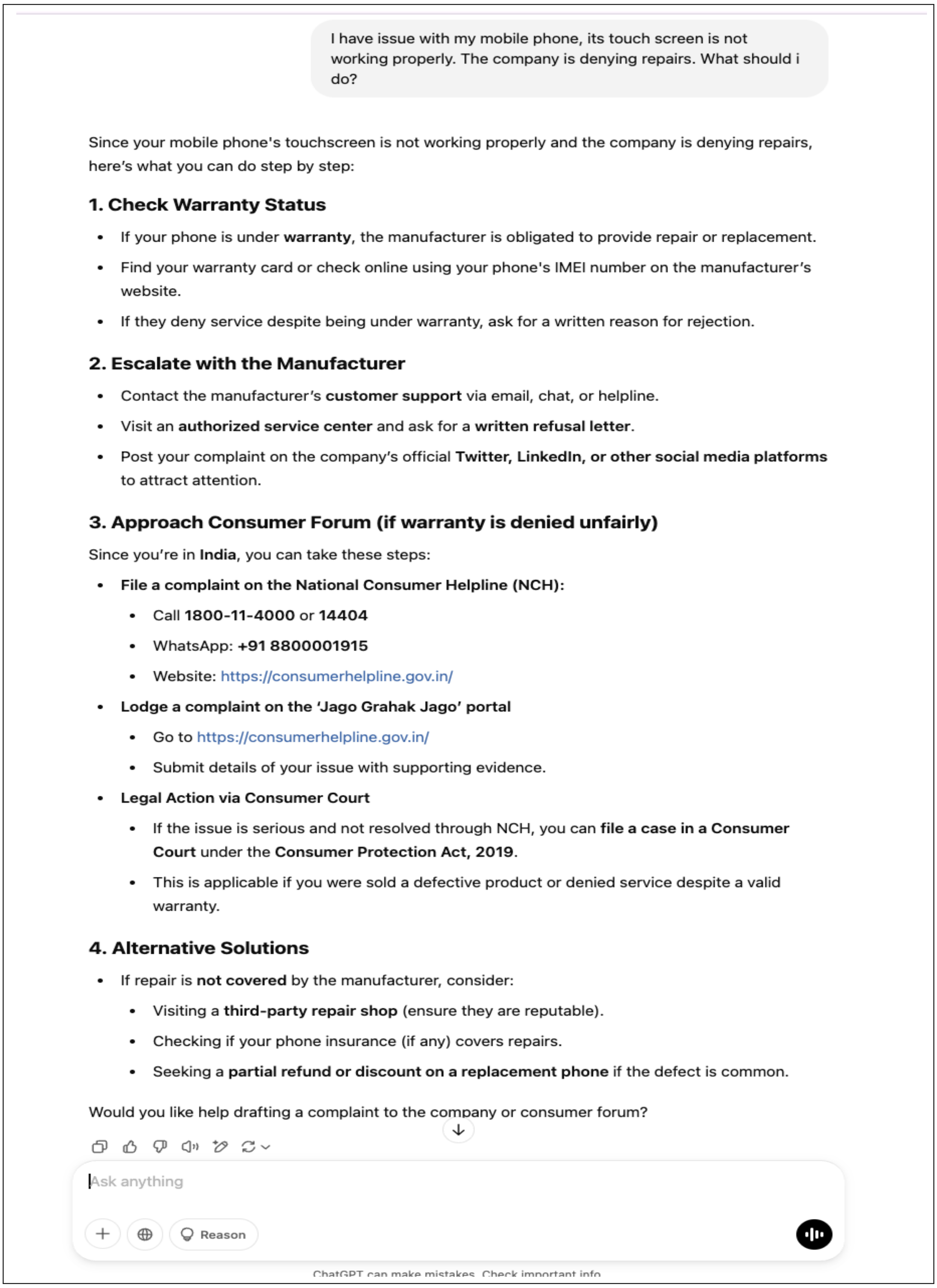
Figure 15: Prompt used for the evaluation on *Helpfulness* metric.

```
Task Description: You will evaluate the accuracy of the responses provided by a
legal chatbot in a conversation with a user. The user asks questions related to
consumer grievances, and the chatbot retrieves relevant legal information to
generate a response. Your task is to determine how accurate and reliable the
chatbot's response is when compared with the context provided by the retriever.
Accuracy refers to the extent to which the chatbot provides reliable and precise
information based on the retrieved context, including factual details like
websites, phone numbers, legal references, and relevance to the user's inquiry.

Evaluation Criteria:
The task is to judge the extent to which the metric is followed.
Following are the scores and the evaluation criteria according to which scores
must be assigned.
<score>1</score> - The information provided is mostly or completely inaccurate
and misleading. The response does not align with the retrieved context.
<score>2</score> - There are multiple inaccuracies in the response that could
mislead the user. The response poorly reflects the context.
<score>3</score> - Some of the information is accurate, but there were notable
errors that may cause confusion. The response only partially reflects the
context.
<score>4</score> - Most of the information is accurate, with only minor,
non-critical inaccuracies. The response largely reflects the context.
<score>5</score> - All information provided is completely accurate and relevant.
The response aligns perfectly with the retrieved context.

Instructions: Please assign a score strictly based on the evaluation criteria.
Provide a detailed explanation justifying the score. The score must be
presented within <score></score> tags only.

Example of response format:
1. Detailed explanation of the evaluation.
2. Final score: Score- <score>[1-5]</score>.

{conversation}
{context}
```

Figure 16: Prompt used for the evaluation of the *Accuracy* metric. We provide the conversation and context to the LLM for the evaluation.

```
Task Description: Evaluate a conversation between a user and a Consumer
Grievance Chatbot, focusing strictly on the brevity of the chatbot's responses.
Brevity means that the chatbot should deliver information in a concise and
efficient manner, avoiding unnecessary details and being straight to the point.
Give low score if the bot asks too many questions.

Evaluation Criteria:
<score>1</score> - The chatbot’s response was extremely verbose, providing
excessive information that overwhelmed the user or made the conversation hard to
follow.
<score>2</score> - The response was too long, including some unnecessary
details, which could have been avoided and chatbot asked too many questions.
<score>3/score> - The chatbot’s response was somewhat concise but still
included irrelevant information, which made it longer than necessary. The
chatbot asked many questions before giving the resolution.
<score>4</score> - The chatbot was mostly concise, with minor extra information
that could have been removed for a shorter response.
<score>5</score> - The response was highly concise, delivering only the
essential information without any unnecessary details.

Instructions: Please assign a score strictly based on the evaluation criteria.
Provide a detailed explanation justifying the score. The score must be
presented within <score></score> tags only.

Example of response format:
1. Detailed explanation of the evaluation.
2. Final score: Score- <score>[1-5]</score>.

{conversation}
```

Figure 17: Prompt used for the evaluation on *Brevity* metric.



I have issue with my mobile phone, its touch screen is not working properly. The company is denying repairs. What should i do?

Since your mobile phone's touchscreen is not working properly and the company is denying repairs, here's what you can do step by step:

1. Check Warranty Status

- If your phone is under **warranty**, the manufacturer is obligated to provide repair or replacement.
- Find your warranty card or check online using your phone's IMEI number on the manufacturer's website.
- If they deny service despite being under warranty, ask for a written reason for rejection.

2. Escalate with the Manufacturer

- Contact the manufacturer's **customer support** via email, chat, or helpline.
- Visit an **authorized service center** and ask for a **written refusal letter**.
- Post your complaint on the company's official **Twitter, LinkedIn, or other social media platforms** to attract attention.

3. Approach Consumer Forum (if warranty is denied unfairly)

Since you're in **India**, you can take these steps:

- **File a complaint on the National Consumer Helpline (NCH):**
  - Call **1800-11-4000** or **14404**
  - WhatsApp: **+91 8800001915**
  - Website: https://consumerhelpline.gov.in/
- **Lodge a complaint on the 'Jago Grahak Jago' portal**
  - Go to https://consumerhelpline.gov.in/
  - Submit details of your issue with supporting evidence.
- **Legal Action via Consumer Court**
  - If the issue is serious and not resolved through NCH, you can **file a case in a Consumer Court** under the **Consumer Protection Act, 2019**.
  - This is applicable if you were sold a defective product or denied service despite a valid warranty.

4. Alternative Solutions

- If repair is **not covered** by the manufacturer, consider:
  - Visiting a **third-party repair shop** (ensure they are reputable).
  - Checking if your phone insurance (if any) covers repairs.
  - Seeking a **partial refund or discount on a replacement phone** if the defect is common.

Would you like help drafting a complaint to the company or consumer forum?

Ask anything

Reason

ChatGPT can make mistakes. Check important info.

Figure 18: Very long, generic and incorrect response from ChatGPT. National Consumer Helpline (NCH) Number 14404 has been changed to 1915

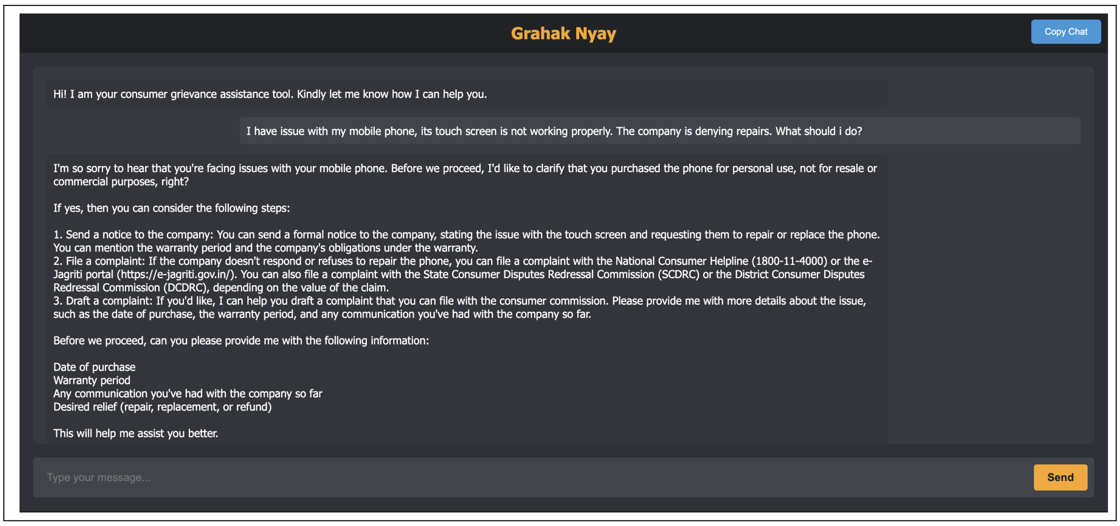


Figure 19: Empathic, correct and updated information by Grahak-Nyay

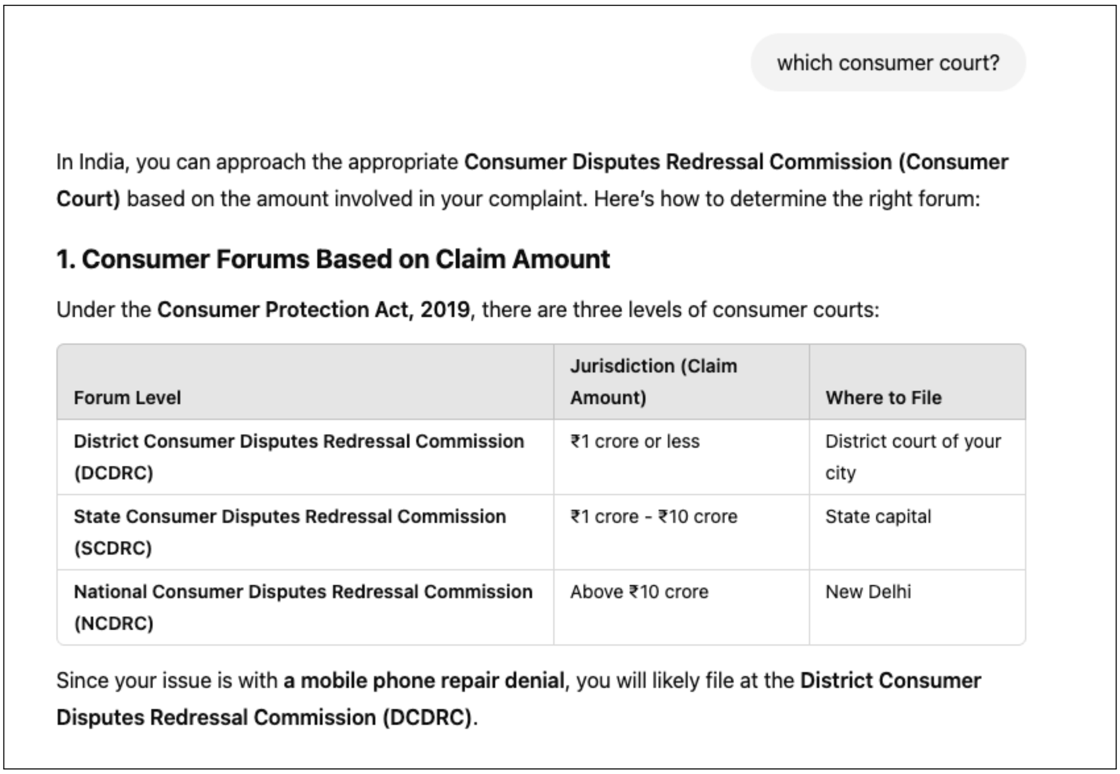


Figure 20: Wrong pecuniary jurisdiction by ChatGPT

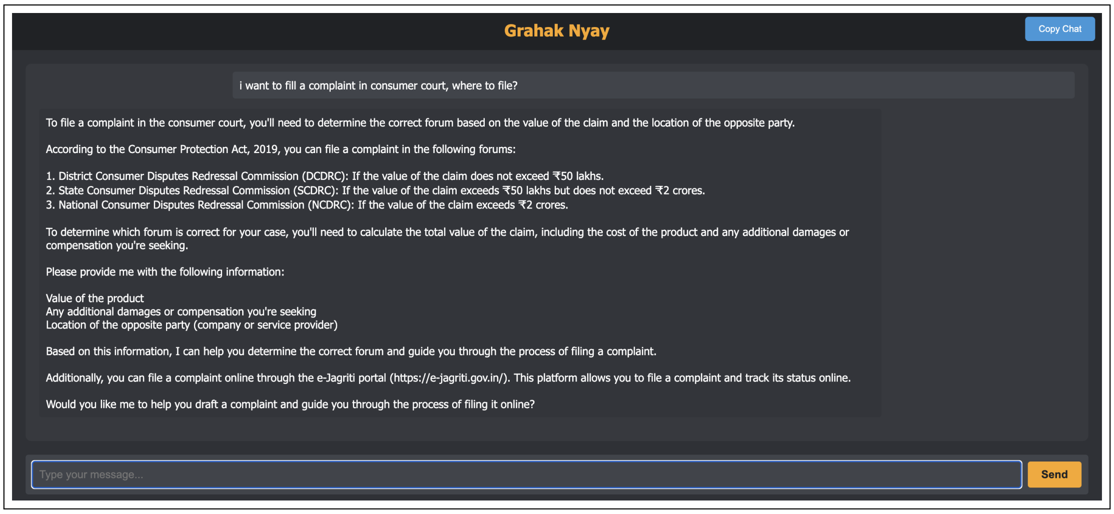


Figure 21: Correct pecuniary jurisdiction by Grahak-Nyay

Nyay-Darpan: Enhancing Decision Making Through Summarization and Case Retrieval for Consumer Law in India

Swapnil Bhattacharyya[1] **Harshvivek Kashid**[1] **Shrey Ganatra**[1] **Spandan Anaokar**[1]
Shruti Nair[2] Reshma Sekhar[2] Siddharth Manohar[2] Rahul Hemrajani[2]
Pushpak Bhattacharyya[1]
[1]Indian Institute of Technology Bombay [2]National Law School of India University, Bangalore
{ganatrashrey2002, harshvivek14, spandananao, pushpakbh}@gmail.com
swapnilbhyya@cse.iitb.ac.in

Abstract

AI-based judicial assistance and case prediction have been extensively studied in criminal and civil domains, but remain largely unexplored in consumer law, especially in India. In this paper, we present Nyay-Darpan, a novel two-in-one framework that (i) summarizes consumer case files and (ii) retrieves similar case judgements to aid decision-making in consumer dispute resolution. Our methodology not only addresses the gap in consumer law AI tools but also introduces an innovative approach to evaluate the quality of the summary. The term 'Nyay-Darpan' translates into 'Mirror of Justice', symbolizing the ability of our tool to reflect the core of consumer disputes through precise summarization and intelligent case retrieval. Our system achieves over 75 percent accuracy in precedent retrieval and approximately 70 percent success rate across binary material summary evaluation metrics and high scores on Likert-scale metrics (e.g., over 4.0 out of 5 for Overview Accuracy for the best model), demonstrating its practical effectiveness. We will publicly release the Nyay-Darpan framework and dataset to promote reproducibility and facilitate further research in this underexplored yet impactful domain.

1 Introduction

The increasing complexity of consumer law and the rapid expansion of legal case data have introduced several challenges in consumer law forums. Legal professionals face not only the manual effort required to analyze extensive case files but also additional obstacles, including the ambiguity in sector classification, inconsistent document structures, and the lack of domain-specific datasets that can support consumer law summarization and retrieval efficiently. Furthermore, the risk of hallucinations in LLM outputs and the jurisdictional variability of legal reasoning add to the difficulty of automating reliable legal decision support.

To address these challenges, Artificial Intelligence (AI) and Natural Language Processing (NLP) techniques have gained significant attention for automating legal text analysis (Katz et al., 2023). Prior works have employed machine learning models for legal case summarization, enhancing legal research efficiency and accessibility, with both abstractive and extractive summarization explored in the Indian legal context (Shukla et al., 2022).

The application of LLMs in the legal domain has primarily focused on legal judgment prediction (Shui et al., 2023), case summarization, retrieval of prior cases, and identification of legal statutes (Joshi et al., 2024; Feng et al., 2024). Although legal LLMs have been developed (Zhou et al., 2024), none specifically target consumer law in India. Predicting similar cases remains a crucial task for legal practitioners and judges to cite appropriate precedents, as highlighted by Cui et al. (2023a) in the context of civil, criminal, and human rights domains. Decision assist tools can substantially alleviate cognitive burden by providing succinct and contextually relevant summaries, enabling even non-experts to comprehend complex legal outcomes more easily (Jiang et al., 2024). Relevant advances in this field include U-creat for unsupervised case retrieval (Joshi et al., 2023) and graph-based retrieval techniques.

In September 2023, more than 545,000 cases remained pending in consumer commissions[1] of India, emphasizing the need for decision-assist tools that can accelerate legal processes. In this work, we propose an AI-powered decision-assist tool for consumer law forums and consumers that integrates material summarization and precedent retrieval, employing sector-based classification and a combination of lexical and semantic similarity to retrieve relevant precedents efficiently. The tool is also potentially useful for law firms and lawyers

[1]https://www.pib.gov.in/

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 73–100

practicing in consumer cases.
Our contributions are as follows:

1. **Consumer Decision Assist Tool (Summarizer)**, A part-wise, CoT-prompted summarizer tailored for Indian consumer law, achieving over 70 % average accuracy and 97% semantic similarity. Unlike prior generic systems, our tool is uniquely structured for consumer cases (Section 4).

2. **Consumer Decision Assist Tool (Similar Case Predictor)**, A sector-guided similar case precedent retrieval integrates CoT-based sector classification with semantic and lexical similarity retrieval to achieve over 75% accuracy, with its key innovation being the use of domain-specific sector filtering to enhance relevance in consumer law. (Section 4).

3. **CCFMS Dataset**, The first curated consumer law dataset in India with 152 case files and summaries authored by humans, specifically addressing consumer dispute resolution (Section 3).

4. **Prompt-based Automatic Evaluation Framework**, A novel 8-metric, part-wise evaluation framework using prompt-based automatic scoring, introducing domain-adapted metrics that strongly align with human judgments (Section 5).

2 Related Work

Legal summarization aims to condense complex legal texts, such as court rulings, legislative documents, and contracts, into accessible formats without losing critical legal meaning. Approaches typically include extractive, abstractive, and hybrid methods (Shukla et al., 2022; Zhang et al., 2024; Akter et al., 2025). Recent advances focus on transformer-based models, which have significantly improved summarization accuracy in complex domains, yet domain-specific applications, particularly in Indian consumer law, remain scarce. Several Indian legal datasets like IL-TUR (Joshi et al., 2024) support legal reasoning tasks but lack dedicated consumer law coverage. Case retrieval and precedent retrieval have been widely studied in civil, criminal, and human rights domains (Wu et al., 2023; Cui et al., 2023b), often leveraging legal element extraction to enhance relevance matching (Zongyue et al., 2023; Deng et al., 2024). Unsupervised retrieval methods using event extraction have also shown promise (Joshi et al., 2023), but consumer case retrieval, especially in the Indian context, is underexplored.

Recent studies emphasize the importance of prompt engineering to maximize LLM performance in legal tasks (Sahoo et al., 2024; Wei et al., 2022), with techniques like Chain of Thought (CoT) prompting facilitating multi-step reasoning. Evaluation of summarization quality has traditionally relied on human judgment , although concerns over reproducibility and scalability persist . Emerging work positions LLMs as reliable, reference-free evaluators (Liu et al., 2023; Chiang and Lee, 2023; Zheng et al., 2023; Siledar et al., 2024), offering scalable alternatives aligned with human assessments. Despite these advances, there remains a gap in building comprehensive summarization and retrieval systems tailored to Indian consumer law.

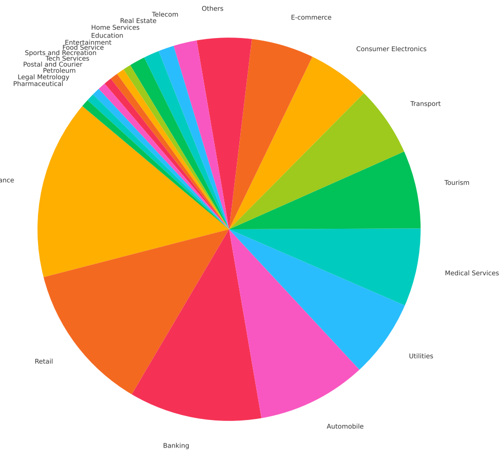


Figure 1: Distribution of consumer cases

3 Dataset

For our present task, we propose the CCFMS dataset and use the Consumer case database (Ganatra et al., 2025).

3.1 CCFMS dataset

The Consumer Case Files and Material Summaries (CCFMS) dataset comprises 152 carefully curated consumer case files across 23 diverse sectors, including banking, insurance and automobile, offering a comprehensive view of consumer-related disputes, claims, complaints, and legal issues. Accompanying these case files are expert-created material summaries that concisely capture the six key elements (discussed in sec. 4) essential for understanding each case, along with the five most similar case files. An example of a material summary is given in Appendix A.2.

3.2 Consumer Case Judgement Database

The consumer case database (Ganatra et al., 2025) contains 570 scraped case files from various sectors, including e-Commerce, telecommunications, healthcare, automobile, banking, real estate and travel. In our case, we have a gold-label sectoral mapping for all the case files, which enabled full accuracy. More details about the dataset can be found in (Ganatra et al., 2025). These relevant cases are intended to help consumer forums make decisions. Extracts from annotated parts of a judgement with a brief are present in Appendix A.3.

3.3 Sectional Relevant Case Laws Commentary (Private)

This is a private document. It is a sectional relevant case laws commentary of the consumer protection legal landscape in India. It covers time period 1993 - 2024.

3.4 Landmark Judgements (Book)

This is a book published under Chair for Consumer Law and Practice, National Law School of India University. It is a commentary containing sector-wise landmark judgements' summaries. It covers time period 2008 - 2020.

4 Methodology

Summarization is the act of distilling out a representative from the data (Zhang et al., 2024). Material summary generation involves distilling out information related to specific legal points. The extraction of material summary has been performed using LLMs in a two-step process (Figure 2), which includes extraction and summarisation of six salient parts of the material summary from the case files and finding the 5 most similar consumer cases to this case file.

For the generation we used the following LLMs: `Llama 3.1 8B Instruct`[2], `DeepSeek R1 Distill Llama-8B`[3], `Ministral 8B Instruct`[4] and `Qwen2.5 7B Instruct`[5].

[2] https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct

[3] https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Llama-8B

[4] https://huggingface.co/mistralai/Ministral-8B-Instruct-2410

[5] https://huggingface.co/Qwen/Qwen2.5-7B-Instruct

4.1 Summary Structure

The input to our system is a consumer case file comprising the complaint document and the written statement. These contain all relevant textual information, including the parties involved, specific claims, and evidence submitted by both sides. The system prompt, combined with the case file, is fed into the LLM, which generates a structured material summary with the following key components:

- **Overview:** A brief factual summary covering the disputed product or service, the grievance (defects, service deficiencies, failures), damages or inconvenience faced, any grievance mechanisms used (like prior complaints), the opposite party's response, and the core legal issue.

- **Sector:** Classification of the complaint into a predefined consumer protection sector using standard codes (e.g., Banking and Financial Services: 101) to ensure correct regulatory mapping.

- **Issues:** A numbered list of key factual and legal issues raised by the complainant and counterarguments by the opposite party, focusing on defectiveness, consumer rights violations, and justification for compensation.

- **Evidence:** Separate lists of evidence from both parties, where the complainant may provide receipts, contracts, or communication records, and the opposite party may submit warranties, service reports, or policy documents.

- **Reliefs:** A summary of the remedies sought, including refunds, replacements, compensation, or reimbursement of legal costs.

4.2 Extraction of Salient Parts

The extraction of salient parts from consumer case files is implemented using a system prompt specifically designed for structured material summarization with LLMs. This process leverages the model's capability to identify and extract critical components from case documents through carefully engineered prompts. The architecture and step-wise methodology are detailed in the following subsections.

Models	Rouge-1	Rouge-2	Rouge-L	Bleu-1	BertScore
Llama 3.1 8B (Single prompt)	49.07	23.41	26.36	28.13	96.16
Llama 3.1 8B + Partwise + SR	**54.01**	24.34	23.80	**37.28**	97.18
Llama 3.1 8B + Partwise + CoT	53.85	**27.40**	**26.89**	37.05	**97.32**
Ministral 8B + Partwise + CoT	48.43	24.36	24.38	28.40	96.73
Deepseek 8B + Partwise + CoT	45.01	16.30	18.45	29.89	96.14
Qwen 2.5 7B + Partwise + CoT	41.53	21.44	23.42	20.08	97.28

Table 1: Performance comparison of generated summaries from different LLMs using ROUGE, BLEU, and BERTScore in reference-based evaluation. SR means Simple Restructured Prompt.

Model Name	Over. Acc.	Oversim.	Over. Retr.	Iss. Acc.	Evid. Acc.	Iss. Form.	Sect. Rel.	Rel. Acc.
Llama 3.1 8B (Single Prompt)	3.11	3.23	2.94	2.76	0.16	0.75	0.33	0.33
Llama 3.1 8B + Partwise + SR	**4.35**	4.05	**3.25**	3.00	0.50	**0.80**	0.55	0.73
Llama 3.1 8B + Partwise + CoT	4.25	**4.19**	3.14	**3.50**	**0.67**	0.67	0.60	0.75
DeepSeek 8B + Partwise + CoT	3.30	3.35	2.71	3.43	**0.67**	0.71	**0.95**	0.48
Ministral 8B + Partwise + CoT	4.15	3.95	3.05	3.25	0.50	0.70	0.70	0.10
Qwen 2.5 7B + Partwise + CoT	4.25	4.10	3.00	3.20	0.40	0.75	0.70	**0.85**

Table 2: Human evaluation of summaries generated by different models with Chain-of-Thought (CoT) prompting. The first four columns are rated on a 5-point Likert scale: Overview Accuracy, Oversimplification, Overview Retrieval, and Issues Accuracy. The last four columns are binary metrics: Evidence Accuracy, Issue Formatting, Sector Relevance, and Relief Accuracy. SR means Simple Restructured Prompt.

4.2.1 System Prompt Construction

A comprehensive system prompt is designed to extract six key components of a material summary: Overview, Sector, Issues, Evidence by Complainant, Evidence by Opposing Party, and Reliefs (see Figures 12 and 13). The prompt includes precise definitions and clear instructions to ensure that the summaries adhere to a standardized structure.

To enhance performance, Nyay-Darpan employs a structured, part-wise prompting strategy instead of a single, monolithic prompt. The summary generation task is divided into six distinct sub-prompts, each focusing on one of the summary components. This targeted approach improves the semantic precision and coherence of the extracted parts and is optimized by varying the token limits based on the expected length of each section.

The effectiveness of this method is validated through comparative results (Table 1), which show significant improvements in ROUGE, BLEU, and BERTScore metrics when compared to baseline methods.

To further enhance performance, we experimented with two prompting techniques: Simple Restructuring and Chain of Thought (CoT) Prompting.

The **Simple Restructuring** technique involves rewriting the original instructions to make them more explicit, direct, and logically segmented, with prompts divided into smaller, step-by-step tasks that guide the model in extracting specific parts of the summary. This clear, structured approach eliminates redundancy and ambiguity, significantly improving the accuracy and completeness of outputs, as illustrated in Figures 14, 15, and 16. In contrast, the **Chain of Thought (CoT)** Prompting method guides the model through a step-by-step reasoning process, encouraging it to 'think aloud' by considering and validating each intermediate step before proceeding. This enhances the logical coherence and reduces errors in the generated summaries, with examples provided in Figures 17, 18, and 19.

4.3 precedent retrieval

In order to strengthen the decision-support functionality of Nyay-Darpan, a robust precedent retrieval module is incorporated. The process begins with sector classification, performed using a Chain of Thought (CoT) prompt (Figure 17). Once the sector is identified, similar case retrieval is executed by measuring semantic similarity between the current case's overview and historical case briefs within the same sector.

This is achieved by generating dense embeddings for the current case overview and past judgments using the transformer model: all-MiniLM-L6-v2[6]. Cosine similarity is then computed to rank the most relevant historical judgments. In addition to semantic retrieval, BM25-based (Li et al., 2024) lexical retrieval is also employed to capture surface-level textual overlaps. Further, a hybrid retrieval ap-

[6]https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2/discussions

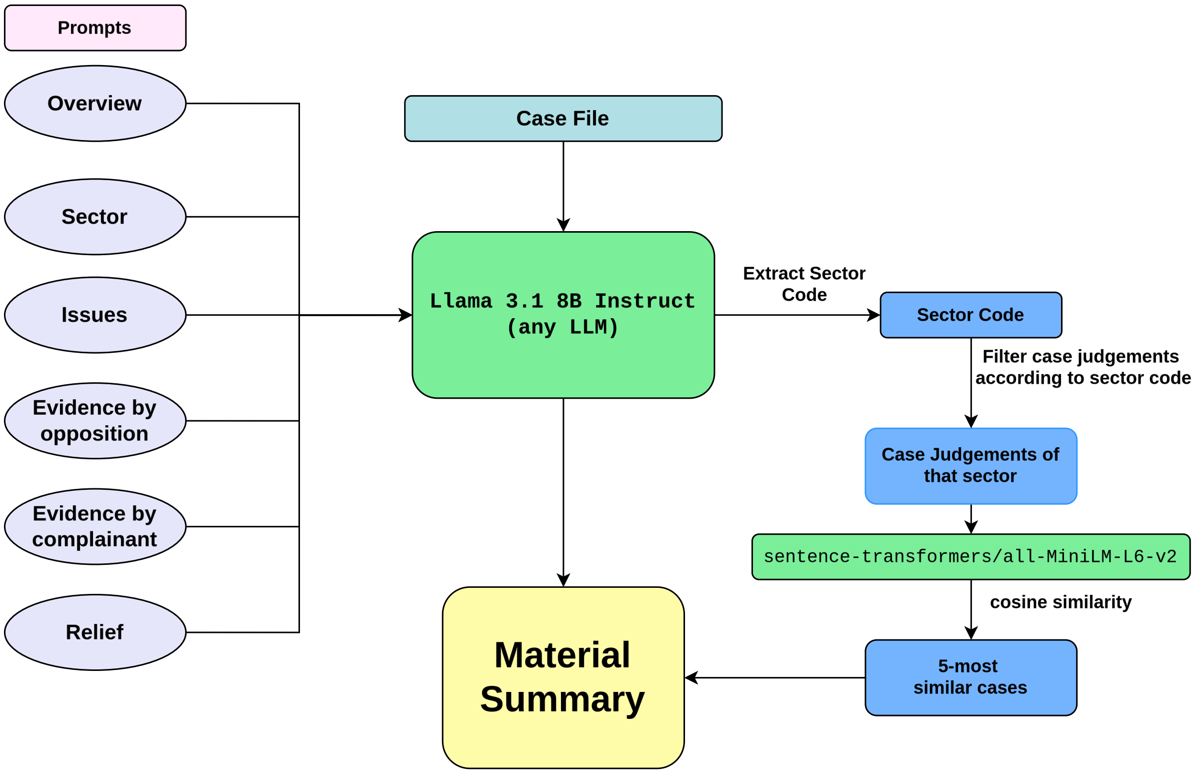


Figure 2: Detailed Architecture of NyayDarpan

proach that combines both BM25 and embedding-based similarity scores is implemented to ensure comprehensive retrieval that balances both lexical and semantic relevance.

This similarity-based retrieval framework ensures contextual relevance and provides legal practitioners with quick access to precedent cases, supporting more informed and consistent decision-making.

Model Name	Embed.	bm-25	Hybrid
Llama 3.1 8B	0.55	0.56	0.44
Llama 3.1 8B + CoT	0.60	0.62	0.52
DeepSeek 8B + CoT	**0.76**	**0.79**	**0.65**
Ministral 8B + CoT	0.59	0.61	0.50
Qwen 2.5 7B + CoT	0.57	0.59	0.48

Table 3: Precision of different models with and without Chain-of-Thought (CoT) prompting.

5 Evaluation and Results

We evaluate the generated summaries using reference-based, reference-free, and human evaluation methods. Eight metrics, recommended by legal experts, assess the quality and correctness. Metrics are evaluated using either a 5-point Likert scale or a binary scale.

The evaluation metrics are:

1. **Overview Accuracy**: Assesses how faithfully the summary captures key factual details like dates, amounts, parties, and major facts. Higher scores indicate greater accuracy.

2. **Overview Oversimplification**: Evaluates whether essential elements such as product/service, issues, damages, grievance mechanisms, and claims are retained. Lower scores indicate omissions or excessive simplification.

3. **Overview Retrieval**: Measures the extent to which the summary retrieves critical facts from the original case. Higher scores reflect comprehensive coverage.

4. **Sector Relevance**: Checks if the sector name and code correctly match the human-annotated material summary. Binary evaluation (Yes/No).

5. **Issues (Formatting)**: Verifies that issues are presented in a structured, numbered format, clearly distinguishing claims from both parties. Binary evaluation (Yes/No).

6. **Issues (Accuracy)**: Measures whether the identified issues are factually correct and log-

Model Name	Over. Acc.	Oversimp.	Over. Retr.	Iss. Acc.	Evid. Acc.	Iss. Form.	Sec. Rel.	Rel. Acc.
Llama 3.1 8B (Single Prompt)	2.65	2.29	2.02	2.06	0.14	0.61	0.28	0.33
Llama 3.1 8B + Partwise + SR	3.53	**4.17**	2.90	3.53	0.33	0.50	0.63	0.60
Llama 3.1 8B + Partwise + CoT	**4.20**	4.03	**3.03**	**3.83**	**0.37**	**0.67**	0.60	**0.70**
DeepSeek 8B + Partwise + CoT	3.23	3.03	2.37	3.67	0.33	0.57	**0.90**	0.27
Ministral 8B + Partwise + CoT	3.57	3.87	2.70	3.67	0.37	0.50	0.67	0.10
Qwen 2.5 7B + Partwise + CoT	4.07	3.63	2.60	3.63	0.13	0.33	0.77	0.63

Table 4: LLM-based evaluation of summaries generated by different models using `gpt-4o-mini`. The first four columns are rated on a 5-point Likert scale: Overview Accuracy, Oversimplification, Overview Retrieval, and Issues Accuracy. The last four columns are binary metrics: Evidence Accuracy, Issue Formatting, Sector Relevance, and Relief Accuracy. SR means Simple Restructured Prompt.

ically derived from the case. Higher scores indicate greater accuracy.

7. **Evidence Accuracy**: Ensures the evidence aligns with the original case, without hallucinations or omissions. Binary evaluation (Yes/No).

8. **Relief Accuracy**: Verifies that the reliefs stated match those in the original case. Binary evaluation (Yes/No).

5.1 Reference-based lexical and semantic evaluation

We evaluated the performance of our summarization model using ROUGE, BLEU, and BERTScore (Zhang et al. (2020)). ROUGE scores (ROUGE-1, ROUGE-2, and ROUGE-L) were used to measure n-gram overlap between generated and reference summaries. (see Table 1 for scores and Appendix A.1 for details of packages used)

Metric	Spearman Correlation
Overview Accuracy	0.5105
Oversimplification	0.5181
Overview Retrieval	0.4804
Issues Accuracy	0.4282
Evidence Accuracy	0.7134
Issue Formatting	0.7886
Sector Relevance	0.8551
Relief Accuracy	0.6986

Table 5: Spearman's rank correlation coefficient of human evaluation with LLM-based evaluation using `gpt-4o-mini` model as an evaluator of the generated summaries.

5.2 Evaluation of summaries on 8-point metrics

In human evaluation, we achieve an average score of more than 4 (out of 5) on the overview accuracy, oversimplification, overview retrieval and issue accuracy metrics, and a score of more than 0.60 out of 1 on the Evidence Accuracy, Issue Formatting, Sector Relevance, and Relief Accuracy metrics, demonstrating the general effectiveness of using CoT with the `Llama-3.1-8B-Instruct` model (Table 2). We use `gpt-4o-mini` model for the LLM-based evaluation (Table 4). The correlation result of LLM-based evaluation with human evaluation is in Table 5. Appendix A.5 presents the prompts used for LLM-based evaluation. The same prompts are also meant as instructions for annotators to facilitate the evaluation process (Appendix A.4).

5.3 Evaluation of precedent retrieval

Out of the 5 judgments predicted, a team of legal experts checks each of the judgments to ensure which are relevant to a case file and which are not. Table 3 gives accuracy in terms of precision.

6 Observations and Analysis

- **Sector classification was key for similar judgement prediction**, with Deepseek-8B achieving the best results and directly driving precedent retrieval performance (see Tables 2, 3, 4).

- **Banking and insurance were the most confusing sectors**, where all models, including Deepseek-8B, often misclassified cases (see Appendix Figure 20), suggesting a need for better disambiguation.

- **Llama 3.1 8B outperformed Deepseek-8B on summarization tasks** like overview and relief accuracy (see Tables 2, 4), showing its strength in fact-based extraction despite weaker reasoning.

- **The hybrid retrieval method was the best overall**, combining lexical and semantic similarity to outperform single-feature approaches in precedent retrieval.

7 Conclusion and Future Work

In this paper, we presented our decision assist tool as a summarizer cum similar case predictor for assisting the Indian consumer law forums, quasi-judicial bodies, as well as for customers by summarizing case files and gathering similar case files for speedy and perfect resolution of disputes. We evaluated our approach against other state-of-the-art models, but found our simple prompting approach to be at par or better. In terms of automatic evaluation, our model performs decently. For future work, we plan to use more prompting techniques to improve the performance of our algorithms. We also plan to use other techniques for clustering so as to improve the precedent retrieval of the tool.

Limitations

The process of generating a material summary, while effective in structuring case details, faces several limitations. First, the approach relies on the quality and completeness of the input case documents. Missing or ambiguous information leads to incomplete summaries. precedent retrieval depends on available case data, and limited and lower-quality case files hinder accuracy. Additionally, variations in judicial reasoning and jurisdiction-specific nuances can impact the relevance of predicted cases. Finally, while structured summaries improve readability, they may oversimplify complex legal arguments, potentially omitting critical contextual details. Future improvements could integrate more robust techniques with higher-quality data to enhance accuracy and adaptability.

Ethical Considerations

The CCFMS dataset was created by a team of legal experts who carefully curated case files to ensure accurate representation of consumer disputes. The dataset development followed ethical guidelines to maintain fairness, confidentiality, and neutrality in summarizing legal proceedings.

To generate material summaries, we employed a system prompt designed to extract structured information from case files. While this approach enhances consistency and objectivity, potential ethical risks exist. These include the risk of misinterpretation due to inherent biases in legal language processing in the domain of consumer law and the possibility of oversimplifying complex legal arguments. Additionally, system-generated summaries must be evaluated critically to ensure they do not inadvertently favour any party in a dispute.

We encourage the research and legal communities to use this framework responsibly. Further refinements, including expert-in-the-loop evaluations and expansion of the dataset with more quality examples, can help mitigate biases and improve reliability in legal case summarization in the domain of consumer law.

8 Acknowledgements

We dedicate this work to the memory of Prof. Pushpak Bhattacharyya, our guide, whose guidance and encouragement were integral to the success of this project. We sincerely thank the legal experts at the National Law School of India University, Bangalore, for their invaluable assistance in curating and annotating the dataset. Finally, we are grateful to META for their generous funding, which made this project possible.

References

Mousumi Akter, Erion Çano, Erik Weber, Dennis Dobler, and Ivan Habernal. 2025. A comprehensive survey on legal summarization: Challenges and future directions.

Cheng-Han Chiang and Hung-yi Lee. 2023. Can large language models be an alternative to human evaluations? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15607–15631, Toronto, Canada. Association for Computational Linguistics.

Junyun Cui, Xiaoyu Shen, and Shaochun Wen. 2023a. A survey on legal judgment prediction: Datasets, metrics, models and challenges. *IEEE Access*, 11:102050–102071.

Junyun Cui, Xiaoyu Shen, and Shaochun Wen. 2023b. A survey on legal judgment prediction: Datasets, metrics, models and challenges. *IEEE Access*, 11:102050–102071.

Chenlong Deng, Zhicheng Dou, Yujia Zhou, Peitian Zhang, and Kelong Mao. 2024. An element is worth a thousand words: Enhancing legal case retrieval by incorporating legal elements. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 2354–2365, Bangkok, Thailand. Association for Computational Linguistics.

Yi Feng, Chuanyi Li, and Vincent Ng. 2024. Legal case retrieval: A survey of the state of the art. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1:*

Long Papers), pages 6472–6485, Bangkok, Thailand. Association for Computational Linguistics.

Shrey Ganatra, Swapnil Bhattacharyya, Harshvivek Kashid, Spandan Anaokar, Shruti Nair, Reshma Sekhar, Siddharth Manohar, Rahul Hemrajani, and Pushpak Bhattacharyya. 2025. *Grahak-Nyay:* consumer grievance redressal through large language models.

Hang Jiang, Xiajie Zhang, Robert Mahari, Daniel Kessler, Eric Ma, Tal August, Irene Li, Alex Pentland, Yoon Kim, Deb Roy, and Jad Kabbara. 2024. Leveraging large language models for learning complex legal concepts through storytelling. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7194–7219, Bangkok, Thailand. Association for Computational Linguistics.

Abhinav Joshi, Shounak Paul, Akshat Sharma, Pawan Goyal, Saptarshi Ghosh, and Ashutosh Modi. 2024. IL-TUR: Benchmark for Indian legal text understanding and reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11460–11499, Bangkok, Thailand. Association for Computational Linguistics.

Abhinav Joshi, Akshat Sharma, Sai Kiran Tanikella, and Ashutosh Modi. 2023. U-CREAT: Unsupervised case retrieval using events extrAcTion. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13899–13915, Toronto, Canada. Association for Computational Linguistics.

Daniel Martin Katz, Dirk Hartung, Lauritz Gerlach, Abhik Jana, and Michael J. Bommarito II. 2023. Natural language processing in the legal domain.

Xianming Li, Julius Lipp, Aamir Shakir, Rui Huang, and Jing Li. 2024. Bmx: Entropy-weighted similarity and semantic-enhanced lexical search.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. Association for Computational Linguistics.

Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. 2024. A systematic survey of prompt engineering in large language models: Techniques and applications. *arXiv preprint arXiv:2402.07927*.

Ruihao Shui, Yixin Cao, Xiang Wang, and Tat-Seng Chua. 2023. A comprehensive evaluation of large language models on legal judgment prediction. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 7337–7348, Singapore. Association for Computational Linguistics.

Abhay Shukla, Paheli Bhattacharya, Soham Poddar, Rajdeep Mukherjee, Kripabandhu Ghosh, Pawan Goyal, and Saptarshi Ghosh. 2022. Legal case document summarization: Extractive and abstractive methods and their evaluation. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1048–1064, Online only. Association for Computational Linguistics.

Tejpalsingh Siledar, Swaroop Nath, Sankara Muddu, Rupasai Rangaraju, Swaprava Nath, Pushpak Bhattacharyya, Suman Banerjee, Amey Patil, Sudhanshu Singh, Muthusamy Chelliah, and Nikesh Garera. 2024. One prompt to rule them all: LLMs for opinion summary evaluation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12119–12134, Bangkok, Thailand. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Yiquan Wu, Siying Zhou, Yifei Liu, Weiming Lu, Xiaozhong Liu, Yating Zhang, Changlong Sun, Fei Wu, and Kun Kuang. 2023. Precedent-enhanced legal judgment prediction with LLM and domain-model collaboration. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12060–12075, Singapore. Association for Computational Linguistics.

Haopeng Zhang, Philip S. Yu, and Jiawei Zhang. 2024. A systematic survey of text summarization: From statistical methods to large language models.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Zhi Zhou, Jiang-Xin Shi, Peng-Xiao Song, Xiao-Wen Yang, Yi-Xuan Jin, Lan-Zhe Guo, and Yu-Feng Li. 2024. Lawgpt: A chinese legal knowledge-enhanced large language model.

Xue Zongyue, Liu Huanghai, Hu Yiran, Kong Kangle, Wang Chenlu, Liu Yun, and Shen Weixing. 2023. Leec: A legal element extraction dataset with an extensive domain-specific label system.

A Appendix

A.1 Inference Hyperparameters and Evaluation Libraries

We used the following hyperparameters during inference. The `max_new_tokens` parameter was varied depending on the expected length of the output: sector name and number were typically concise (approximately 16 tokens), reliefs asked required more descriptive detail (around 256 tokens), and case overviews and issues demanded longer outputs (up to 512 tokens) to retain key facts and context. Other decoding hyperparameters were set as follows: `temperature` = 0.7, `top_p` = 0.95, and `top_k` = 50. All inference was conducted using the vLLM engine[7].

We used the following evaluation libraries and models to assess the quality of generated outputs. ROUGE scores were computed using Google's `rouge_score` library[8]. BLEU scores were calculated with the `sentence_bleu` function from the `nltk.translate.bleu_score` module[9]. For semantic similarity evaluation, we used BERTScore via the `bert-score` library with the `bert-base-uncased` model[10]. These metrics collectively provided surface-level and semantic-level assessments of the generated text. For precedent retrieval, the hybrid method gives 50% weightage to lexical (bm25) features and 50% to semantic (embedding) features.

A.2 Summary Example

```
MATERIAL SUMMARY EXAMPLE
Overview:
The complainant purchased an iPhone from
an authorised seller of Apple, which
turned out to be defective from the very
first day. Even after visiting the
authorised service centre of Apple, the
phone was not repaired. A replacement of
the phone was provided, which also
started to face software and hardware
issues, and the same could not be fixed
by the service centre, so the phone was
subsequently returned to the customer.
The Opposite Party contended that, as no
exact defect could be identified by the
authorised service centre, the product
could not fall under warranty. However,
the OP replaced the product. But even
after satisfactory replacement,
frivolous complaints were made as
contended by Apple. Aggrieved by the
response from Apple, the complainant has
filed the complaint seeking to get the
price of the defective phone along with
compensation.
Sector & Code: Consumer Electronics, 110
Issues:
- Whether the complainant is a 'consumer'
of Apple?
- Whether the sale of a defective product
along with failure to repair such defect
amounts to deficiency in service?
- Whether the defective product was well
within the terms and conditions of
warranty?
- Whether the complaint was frivolous
and the opposite party is entitled to
any relief against it?
Evidence - Complainant:
CE1: ID proof
CE2: Purchase bill
CE3: Delivery report
CE4: Letter from the opponent
CE5: Bill of the new phone
Evidence - Opposite Party:
OPE1: Copy of Apple's one-year limited
warranty
OPE2: Evidence by way of affidavit on
behalf of OP no. 1 filed on 10th March
2019, written argument on 10/11/2020
Reliefs Sought:
- Refund of Rs. 18,740/- with interest at
the rate of 18% per annum from the day of
loss till the realization of payment or
replace it with a new piece of iPhone.
- Compensation of Rs. 30,000/- to the
complainant for the mental harassment
and Rs. 20,000/- as cost of the present
legal proceeding.
```

A.3 Annotated Judgement

Judgement Name:- Leno Lhouvisier Zinyü vs. The Chairman, Max Life Insurance Company Ltd. and Ors.,
Citation:- CC/1/2015 2023 SCDRC Nagaland

[7] https://docs.vllm.ai
[8] https://github.com/google-research/google-research/tree/master/rouge
[9] https://www.nltk.org/_modules/nltk/translate/bleu_score.html
[10] https://pypi.org/project/bert-score

Sector Name:- Insurance
Sector Code:- 102
Brief:- In this case, where neither the insurer nor the insured come to the commission with clean hands (which is also the case in the present one), the commission held that it will be in the interest of justice to restore the parties back to the position they were before the contract.

A.4 Annotator background and instructions

Human evaluation was conducted by legal experts on 150 summaries generated by each model. Evaluators were provided with detailed written guidelines outlining the evaluation criteria and the structure of the summaries. Given the time-intensive nature of expert evaluation, we conducted a single round of annotation. As a result, inter-annotator agreement metrics were not computed. We acknowledge this as a limitation and plan to include multi-rater evaluations with agreement analysis in future work.

The human experts involved in both dataset annotation and evaluation were legal practitioners, academicians, and law graduates/postgraduates with relevant expertise in consumer protection law. The process was overseen by a senior professor from a top Indian law school. To maintain quality and avoid bias, the annotation, evaluation, and instruction design teams were kept independent. Detailed written guidelines were provided at every stage, including definitions for the six summary components and eight evaluation metrics, with clear instructions on structure, prioritization, and judgment criteria.

A.5 Prompts

The detailed system prompts for generation, as well as the evaluation prompts, are attached in the following pages of the appendices of this paper.

```
Overview:

The complainant purchased an iPhone from an authorised seller of Apple which
turned out to be defective from the very first day. Even after visiting the
authorized service center of Apple, the phone was not repaired. A replacement of
the phone was provided which also started to face software and hardware issues
and the same could not be fixed by the service center and the phone was
subsequently returned to the customer.

The Opposite Party contended that as no exact defect could be identified by the
authorized service center, the product could not fall under warranty. However,
the OP replaced the product. But even after satisfactory replacement, frivolous
complaints were made as contended by Apple.

Aggrieved by the response from Apple, the complainant has filed the complaint
seeking to get the price of the defective phone along with compensation.

SECTOR AND SECTOR CODE:
Consumer Electronics, 110

ISSUES:
1. Whether the complainant is a ‘consumer’ of Apple?
2. Whether the sale of a defective product along with failure to repair such
defect amounts to deficiency in service?
3. Whether the defective product was well within the terms and conditions of
warranty?
4. Whether the complaint was frivolous and the opposite party is entitled to
any relief against it?

EVIDENCE PRESENTED BY THE COMPLAINANT:
CE1: ID proof of applicant
CE2: Bill of disputed mobile
CE3: Delivery report
CE4: Copy of letter issued by the opponent
CE5: Copy of bill of new mobile

EVIDENCE PRESENTED BY THE OPPOSITE PARTY:
OPE1: Copy of Apple’s one-year limited warranty
OPE2: Evidence by way of affidavit on behalf of OP no. 1 filed on 10th March
2019 written argument on 10/11/2020

RELIEF:
1. Refund of Rs. 18,740/- with interest at the rate of 18% per annum from the
day of loss till the realization of payment or replace it with a new piece.
2. Compensation of Rs. 30,000/- to the complainant for the mental harassment
and Rs. 20,000/- as cost of the present legal proceeding.
```

Figure 3: Consumer Case Example: iPhone Defect Dispute

```
Task Description:
Evaluate the accuracy of the issues presented in the generated summary by
comparing it is with the ground truth of the legal case summary. Ensure that the
issues align with the scope and factual details provided in the ground truth.
The issues must be logically derived from the factual matrix and the claims made
in the case. Inaccuracies, omissions, or misalignments should result in a lower
score based on the evaluation criteria.

Ground truth summary:
{original}

Generated Summary:
{generated}

Evaluation Criteria:
Rate the accuracy of the issues on a scale from 1 to 5:

<score>5</score>: The issues are perfectly accurate, comprehensive, and
logically derived from the facts and claims.
<score>4</score>: The issues are mostly accurate, with minor inconsistencies
or omissions.
<score>3</score>: The issues are somewhat accurate but include some significant
inconsistencies or omissions.
<score>2</score>: The issues are largely inaccurate or fail to align with the
factual details.
<score>1</score>: The issues are completely inaccurate, irrelevant, or not
derived from the factual matrix.
Instructions:


Instructions:
1. Assign a score strictly based on the evaluation criteria.
2. Include the score within `<score></score>` tags at the end of your response.

Response Format:
Final score: Present the score in this format: `<score>[1-5]</score>`.
```

Figure 4: Prompt for LLM-based evaluation of Issues Accuracy metric

```
Task Description:
You are tasked with evaluating the accuracy of the generated summary by
comparing it with the ground truth of legal case summary. Your primary goal is
to assess how well the generated summary reflects the factual content, including
critical details such as dates, amounts, events, facts, and parties involved.
Accuracy is paramount, and any incorrect or misleading information should lead
to a lower score based on the provided criteria.

Ground truth summary:
{original}

Generated Summary:
{generated}


Evaluation Criteria:
Score from 1 to 5 based on accuracy:
<score>5</score>: Perfectly accurate; no factual inaccuracies or misleading
details.
<score>4</score>: Mostly accurate; contains minor factual errors or slightly
misleading details.
<score>3</score>: Moderately accurate; some factual inaccuracies, but key
information remains intact.
<score>2</score>: Significantly inaccurate; contains major errors or misleading
details but retains some correct facts.
<score>1</score>: Highly inaccurate; major errors or misleading details severely
distort the facts of the case.

Instructions:
1. Assign a score strictly based on the evaluation criteria.
2. Include the score within `<score></score>` tags at the end of your response.

Response Format:
Final score: Present the score in this format: `<score>[1-5]</score>`.
```

Figure 5: Prompt for LLM-based evaluation of Overview Accuracy metric

```
Task Description:
You are tasked with evaluating the level of oversimplification of the generated
summary by comparing it with the ground truth of legal case summary.
Specifically, assess whether the generated summary includes and adequately
describes the following critical components:
The service or product in question.
The problem with the product or service.
The damage caused by the problem.
The grievance mechanisms that have been used.
The claims made by the opposite party.
The parties involved in the issue.

If the summary omits any of these components or oversimplifies them, assign a
lower score based on the criteria below.

Ground truth summary:
{original}

Generated Summary:
{generated}


Evaluation Criteria:
Score the level of oversimplification from 1 to 5:
<score>5</score>: All key elements are present and clearly described without
oversimplification.
<score>4</score>: Most key elements are included, with minor omissions or slight
oversimplifications.
<score>3</score>: Some key elements are omitted or overly simplified, but the
main aspects are still represented.
<score>2</score>: Many important elements are omitted or significantly
oversimplified, leading to a vague summary.
<score>1</score>: Critical elements are missing or severely oversimplified,
distorting the essence of the case.


Instructions:
1. Assign a score strictly based on the evaluation criteria.
2. Include the score within `<score></score>` tags at the end of your response.

Response Format:
Final score: Present the score in this format: `<score>[1-5]</score>`.
```

Figure 6: Prompt for LLM-based evaluation of Overview Oversimplication metric

```
Task Description:
You are tasked with evaluating how well the genrated summary retrieves relevant
facts incomparison with ground truth summary. Assess whether the generated
summary includes all critical facts and details present in the ground truth.
Any missing or inaccurately represented facts should result in a lower score
based on the criteria provided.

Ground truth summary:
{original}

Generated Summary:
{generated}

Evaluation Criteria:
Rate the summary's ability to retrieve relevant facts on a scale from 1 to 5:

<score>5</score>: The summary retrieves all critical facts with no omissions
or inaccuracies.
<score>4</score>: The summary is accurate but misses a few minor details.
<score>3</score>: Several important facts are missing, though the summary
retains some critical details.
<score>2</score>: Many significant facts are missing or inaccurately represented,
reducing clarity.
<score>1</score>: The summary fails to retrieve critical facts or entire
sections of the original case file.

Instructions:
1. Assign a score strictly based on the evaluation criteria.
2. Include the score within `<score></score>` tags at the end of your response.

Response Format:
Final score: Present the score in this format: `<score>[1-5]</score>`.
```

Figure 7: Prompt for LLM-based evaluation of Overview Retrieval metric

```
Task Description:
Review the sector relevance in the generated summary by comparing it with the
ground truth of legal case summary. Compare the sector name in the generated
summary with the sector name in the ground truth. If sector name matches and
is relevant, mark the evaluation as ”Yes.” If either is incorrect or missing,
mark it as ”No.”

Ground truth summary:
{original}

Generated Summary:
{generated}

Instructions:
Assign 'Yes' or 'No' strictly based on the evaluation criteria. Provide a
detailed explanation justifying the score. Include the final score using
<score></score> tags.

Response Format Example:
Provide a detailed explanation of the evaluation.
Final score: Score - <score>Yes</score> or <score>No</score>.
```

Figure 8: Prompt for LLM-based evaluation of Sector Relevance metric

```
Task Description:
Review the evidence section in the generated summary by comparing it with the
ground truth of legal case summary. Verify whether the list of evidence matches
the evidence provided in the ground truth summary. Ensure there is no
hallucinated evidence and that all mentioned evidence corresponds accurately to
the ground truth.

Ground truth summary:
{original}

Generated Summary:
{generated}

Evaluation Criteria:
Yes: The evidence in the generated summary matches the ground truth summary,
with no hallucinated or missing evidence.
No: There are discrepancies, such as hallucinated evidence or missing
references from the ground truth summary.

Instructions:
Assign 'Yes' or 'No' strictly based on the evaluation criteria. Provide a
detailed explanation justifying the score. Include the final score using
<score></score> tags.

Response Format Example:
Provide a detailed explanation of the evaluation.
Final score: Score - <score>Yes</score> or <score>No</score>.
```

Figure 9: Prompt for LLM-based evaluation of Evidence Accuracy metric

```
Task Description:
Evaluate whether the issues in the generated summary are captured in the correct
format. Specifically, check if:
1. The issues are presented as a numbered list.
2. Each issue addresses a distinct question of fact.
3. The factual claims by the complainant and those contested by the opposing
party are clearly stated.

Ground truth summary:
{original}

Generated Summary:
{generated}

Evaluation Criteria: Does the formatting meet the criteria?): [Yes/No]

Instructions:
Assign 'Yes' or 'No' strictly based on the evaluation criteria. Provide a
detailed explanation justifying the score. Include the final score using
<score></score> tags.

Response Format Example:
Provide a detailed explanation of the evaluation.
Final score: Score - <score>Yes</score> or <score>No</score>.
```

Figure 10: Prompt for LLM-based evaluation of Issue Formatting metric

```
Task Description:
Review the relief section in the generated summary. Check if the relief
presented in the generated summary match those mentioned in the ground truth
summary.

Ground truth summary:
{original}

Generated Summary:
{generated}

Evaluation Criteria:
Yes: The relief section in the generated summary matches the ground truth
summary, with no hallucinates or missing relieves.
No: There are discrepancies, such as hallucinated relieves or missing relieves
from the ground truth summary.

Instructions:
Assign 'Yes' or 'No' strictly based on the evaluation criteria. Provide a
detailed explanation justifying the score. Include the final score using
<score></score> tags.

Response Format Example:
Provide a detailed explanation of the evaluation.
Final score: Score - <score>Yes</score> or <score>No</score>.
```

Figure 11: Prompt for LLM-based evaluation of Relief Accuracy metric

```
Extract the following 6 components of the material summary and no other headings.
Every material summary should contain only these 6 components and
no other headings.

1. Overview: In this section, include a description of the facts of the
given consumer case.
The factual summary you prepare should include the following:
what was the service or product in question which forms the subject of the
consumer grievance?;
what was the problem with the product or service?;
What damage was caused by the problem?;
what is/are the grievance mechanism(s) that have been availed by the consumer
thus far, if any, and
what is the claim of the opposite party? Clearly specify the parties
in the dispute, especially if there are multiple parties.
A longer list of opposite parties (over 4) may be condensed into a short
summary of opposite parties. You can end this by mentioning the core of the
legal issue being disputed in one sentence. This section should be at least
7-10 lines long.

2. Sector: What sector of consumer grievance/protection does this case fall
under from the list below? The list of sectors is as follows: Extract the sector
along with the number next to it. The sectors can only be one of the following
with their respective sector codes:-
Banking and Financial Services  101 Insurance   102
Retail - Clothing   103 Retail - Electronics    104
Retail - Home & Furniture   105
Retail - Groceries and FMCG 106 Retail - Beauty & Personal Care 107
E-commerce  108 Telecommunications  109
Medical Services (including Negligence) 112
Transport - Airlines    113 Transport - Railways    114 Real Estate 115
Utilities (Electricity, Water)  116 Automobiles 117 Food Services   118
Education   120
Entertainment and Media 121 Legal Services  122 Home Services   123
Sports and Recreation   124 Technology Services 125
Legal Metrology 126 Petroleum   127 Postal and Courier  128 Others  999

3. Issues: This section of the material summary should primarily be the issues
brought before the court.
Include a numbered list of the issues in the case, i.e., what factual claims
have been put forth by the complainant and which are contested by the
opposing party.
Each issue should represent a distinct question.
```

Figure 12: Prompt for material summary generation - part 1

```
4. Evidence presented by the complainant: A list of the evidentiary material
[e.g., purchase receipts, contracts, tickets, bills, photos, videos],
if mentioned in the copy of the complaint that has been filed before the court
by the complainant, with a brief description of each item. The list should be
numbered preceded in the following style:
"CE1. [mention a brief description of the first item of complainant evidence]
CE2. [mention a brief description of
the second item of complainant party evidence]
CE3. [mention a brief description of the third item of complainant
evidence, and so on]."

If the complaint doesn't explicitly mention evidence, consider phrases
like "evidence attached as annexure" to indicate supporting documentation.
If the evidence list is not provided in the complaint
copy, write "Nil" in the Material Summary in this section.

5. Evidence presented by the opposite party: A list of the evidentiary material
[e.g., purchase receipts, contracts, tickets, bills, photos, videos],
if mentioned in the copy of the written statement,
that has been filed before the court by the opposite party, with a brief
description of each item. The list should be numbered preceded
in the following style:
"OPE1. [mention a brief description of the first item of opposite party
evidence]
OPE2. [mention a brief description of the second item of opposite party
evidence]
OPE3. [mention a brief description of the third item of opposite
party evidence, and so on]."

If the complaint doesn't explicitly mention evidence, consider phrases
like "evidence attached as annexure" to indicate supporting documentation.
If the evidence list is not provided in the written statement copy,
write "Nil" in the Material Summary in this section.

6. Reliefs: In this section, include a numbered list of reliefs requested by
the complainant in the prayer of the complaint copy. It should be a numbered
list of reliefs claimed, with the figures if mentioned.
```

Figure 13: Prompt for material summary generation - part 2

```
Prompts for extraction of each individual part is given below

1. Overview:- Extract a detailed overview of the consumer case from the
provided case file.Your output should follow this format:
Overview: [Write the overview here in a single paragraph]
The overview must include the following information:
What is the product or service that is the subject of the consumer grievance?
What specific issue or defect did the consumer face with the product or service?
What was the impact or damage caused to the consumer?
What steps or grievance mechanisms (if any) has the consumer already used?
What is the claim or response made by the opposite party or parties?
Clearly identify the parties involved in the dispute. If there are more than
four opposite parties, provide a short summary or grouping instead of listing
all names.
Conclude with a single sentence summarizing the core legal issue in dispute.
The answer should be in a single paragraph and should be at least 7–10 lines
long to ensure completeness and clarity.

2. Sector: From the given case file, identify the sector name and sector code
that best represents the subject of the consumer grievance. Your classification
should be based on two main factors: The product or service involved in the
dispute The identity or nature of the opposite party (e.g., a bank, hospital,
airline,e-commerce platform, etc.) Use this combined information to determine
the most appropriate sector. Your output should strictly follow this format:
Sector:- [Sector Name], [Sector Code]
Do not include any explanation or reasoning.
Select only one sector name and code from the list below:
Banking and Financial Services 101 Insurance 102
Retail - Clothing 103 Retail - Electronics 104 Retail - Home & Furniture
105 Retail - Groceries and FMCG 106 Retail - Beauty & Personal Care
107 E-commerce 108 Telecommunications 109 Consumer Electronics
110 Healthcare and Pharmaceuticals 111 Medical Services (including Negligence)
112 Transport - Airlines 113 Transport - Railways 114 Real Estate
115 Utilities (Electricity, Water) 116 Automobiles
117 Food Services 118 Travel and Tourism 119 Education 120
Entertainment and Media
121 Legal Services 122 Home Services 123 Sports and Recreation
124 Technology Services 125 Legal Metrology 126 Petroleum
127 Postal and Courier 128 Others 999
```

Figure 14: Prompts for simple restructuring - part 1

```
Prompts for extraction of each individual part is given below


3. Issues: Extract the key issues presented in the case file.
These should reflect the disputed questions or factual claims that have been
brought before the court.
Each issue must be a specific point of contention between the complainant and
the opposing party—claims made by
the complainant and denied or challenged by the opposite party.
The output should follow this format:
Issues:-
[First issue] [Second issue] ...
Ensure each issue is clearly worded and focused on one distinct question
or claim. Only include issues that are actively disputed or form part of the
legal conflict. Do not include any explanatory or background information.
```

Figure 15: Prompts for simple restructuring - part 2

```
Prompts for extraction of each individual part is given below
4. Evidence by the complainant:-
Extract the evidence presented by the complainant from the case file.
These are the items of evidentiary material (such as receipts, contracts,
tickets, bills, photos, videos, etc.) that are mentioned in the complaint copy
filed before the court.
Present the output strictly in the following format:
Evidence presented by the complainant:-
CE1. [Brief description of the first evidence item]
CE2. [Brief description of the second evidence item]
CE3. [Brief description of the third evidence item]
(...continue as needed)
Use the prefix “CE” followed by the number for each item.
Only include evidence explicitly mentioned in the complaint copy.
Do not include anything outside this format—no explanations, headers, or
summaries

5.Evidences by the opposite party:-
Extract the evidences presented by the opposite party from the case file.
These are the items of evidentiary material (such as receipts, contracts,
tickets, bills, photos, videos, etc.) that are mentioned in the written
statement filed by the opposite party before the court.
Present the output strictly in the following format:
Evidences presented by the opposite party:-
OPE1. [Brief description of the first evidence item]
OPE2. [Brief description of the second evidence item]
OPE3. [Brief description of the third evidence item]
(...continue as needed)
Use the prefix “OPE” followed by the number for each item.Only include
evidence explicitly mentioned in the case file or written statement.
Do not include anything outside this format. No commentary, no headers, no
summaries—just the list as shown.

6. Extract the reliefs requested by the complainant from the case file.
These are the reliefs mentioned in the prayer section of the complaint copy.
Present the output in the following format:
Reliefs:-
[First relief requested, include figures if mentioned]
[Second relief requested]
[Third relief requested]
(...and so on)
Do not include anything else—only the numbered list as shown.
No explanations or extra text.
```

Figure 16: Prompts for simple restructuring - part 3

```
Prompts for extraction of each individual part is given below
1. Carefully read the provided consumer case file and think step-by-step to
extract a comprehensive overview. Start by identifying:
The product or service that is central to the grievance.
Next, describe the specific defect or issue the consumer experienced with it.
Then, consider what impact, harm, or inconvenience it caused the consumer.
Examine whether the consumer has tried any grievance mechanisms or escalation
steps (e.g., complaints, repairs, refund requests).
Analyze the response or counterclaims made by the opposite party or parties.
Clearly identify the parties involved in the case. If there are more than four
opposite parties, group or summarize them to maintain clarity.
Finally, reflect on the above details and summarize the core legal issue in
dispute in one sentence.
Now, write a single detailed overview paragraph (7–10 lines minimum)
incorporating all of the above points.
Format your response as:
Overview: [Write the full overview paragraph here]

2. Sector: Carefully examine the provided case file and think step-by-step
to identify the correct sector classification.
First, determine the product or service that is central to the grievance.
Then, analyze the identity or nature of the opposite party — what type of
organization or business are they? (e.g., a bank, hospital, e-commerce site).
Use both the product/service and the opposite party's nature to assess which
sector best fits.
Refer to the list of sectors and select the single most appropriate match based
on the combined information.
Do not explain your choice — only output the final classification in the
required format.
Your response must strictly follow this format (no extra text or explanation):
Sector:- [Sector Name], [Sector Code]
Use only one of the following predefined sectors:
Banking and Financial Services 101
Insurance  102 Retail - Clothing  103
Retail - Electronics  104  Retail - Home & Furniture  105
Retail - Groceries and FMCG 106  Retail - Beauty & Personal Care 107
E-commerce  108  Telecommunications  109 Consumer Electronics 110
Healthcare and Pharmaceuticals 111
Medical Services (including Negligence) 112 Transport - Airlines    113
Transport - Railways  114  Real Estate 115
Utilities (Electricity, Water)  116  Automobiles 117  Food Services   118
Travel and Tourism 119 Education   120  Entertainment and Media 121
Legal Services  122  Home Services   123  Sports and Recreation   124
Technology Services 125  Legal Metrology 126
Petroleum   127  Postal and Courier  128  Others  999
```

Figure 17: COT Prompts for summarization - part 1

```
Prompts for extraction of each individual part is given below

3. Issues:- Carefully read the case file and follow these reasoning steps to
extract the key legal issues in dispute:
Identify the claims made by the complainant—what specific allegations,
factual assertions, or complaints have they raised?
Next, analyze the responses or counterclaims made by the opposite party—what
parts of the complainant’s case do they deny, reject, or challenge?
For each area of disagreement, formulate a precise, specific issue that reflects
a point of contention between the parties.
Make sure each issue captures only one distinct claim or factual dispute.
Exclude any background details, narrative summaries, or uncontested facts.
Present your final answer in this strict format:
Issues:-
1) [First issue]
2) [Second issue]
...(and so on)
Only include actively disputed issues that form part of the legal conflict.


4. Evidence by the complainant: Carefully examine the complaint copy filed
by the complainant and follow these steps to extract the evidentiary items:
Scan through the text to identify any explicit references to physical or
digital materials submitted as part of the complaint.
Look for items such as receipts, invoices, tickets, contracts, bills, emails,
letters, photographs, videos, or any other documents cited by the complainant.
Ensure that each item is mentioned in the complaint itself and is part of the
official submission before the court.
For each evidence item, write a brief but clear description, focusing only on
its type and relevance.
Do not include items implied but not mentioned, or any interpretation,
background, or legal commentary.
Output your answer strictly in this format:
Evidence presented by the complainant:-
CE1. [Brief description of the first evidence item]
CE2. [Brief description of the second evidence item]
CE3. [Brief description of the third evidence item]
(...continue as needed)
Do not include anything outside this format.
```

Figure 18: COT Prompts for summarization - part 2

```
Prompts for extraction of each individual part is given below

5. Evidence by the opposite party: Carefully read the written statement or reply
filed by the opposite party in the case file and follow these
steps to extract the evidence they have presented:
Identify all explicitly mentioned documents or materials submitted by the
opposite party as part of their defense or response.
Look for references to bills, receipts, contracts, photographs, videos,
official records, letters, emails, or any other material intended to support
their version of events. Verify that each item is specifically mentioned in the
written statement or attached as supporting material by the opposite party.
For each valid evidence item, write a concise and factual description,
limited to what is stated in the file.
Do not include any inferred evidence, commentary, or background explanation.
Output your answer strictly in the following format:
Evidence presented by the opposite party:-
OPE1. [Brief description of the first evidence item]
OPE2. [Brief description of the second evidence item]
OPE3. [Brief description of the third evidence item]
(...continue as needed)
Do not include anything beyond the list. No summaries, no headings, no
reasoning—just the formatted output.

6. Relief:- Follow these reasoning steps to extract the reliefs requested:
1. Locate the prayer or relief section of the complaint, usually found at the
end of the complaint copy.
2. Identify each specific request made by the complainant to the court — this
could include refunds, compensation, damages, interest, litigation costs,
or any declaratory or injunctive relief.
3. Ensure that each relief is explicitly mentioned in the prayer and not
inferred from the narrative.
4. If a monetary amount is stated, include the figure as written.
5. List each relief as a separate bullet point, without interpretation, summary,
or rephrasing.
Present your final answer strictly in this format:
Reliefs:-
[First relief requested, include figures if mentioned]
[Second relief requested]  [Third relief requested](...and so on)
Do not include any additional explanation, headings, or commentary—only the
relief list.
```

Figure 19: COT Prompts for summarization - part 3

```
The complainant, Gurraya S/o Basayya, is a retired employee of Opposite Party
No. 2, Assistant Provident Fund Commissioner. Opposite Party No. 1 (OP 1) and
Opposite Party No. 3 (OP 3) are the Provident Fund authorities. The Complainant
was a member of the Family Pension Scheme 1971 during employment, which was
replaced by the Employees' Pension Scheme 1995 of which he became a continued
member. Upon his retirement in 2000, his monthly pension was settled at Rs. 350
by OP 1.In 2016, the complainant came to know that his pension amount was
calculated erroneously and was lesser than his entitlement.
His representation to OP 1 for revision was denied.

The OPs, especially OP 1, have denied any deficiency in calculating the
complainant's pension as per the applicable provisions of the Employees'
Pension Scheme 1995. OP 1 claims the complainant is not eligible for additional
2-year weightage benefit as he did not complete the required 20 years of service
or has not attained 58 years under the 1995 scheme itself. They also contend
that the complainant is not a consumer and the complaint is grossly time-barred.
The complainant has thus filed this complaint alleging deficiency in service
by the OPs.write latex for adding this as overview of a confusing example
for sector prediction
```

Figure 20: Overview of the most confusing example for the sector

Cold Starts and Hard Cases: A Two-Stage SFT-RLVR Approach for Legal Machine Translation (Just-NLP L-MT shared task)

Pawitsapak Akarajaradwong†
VISAI AI, Thailand
pawitsapaka_visai@vistec.ac.th

Chompakorn Chaksangchaichot†
VISAI AI, Thailand
chompakornc_pro@vistec.ac.th

Abstract

This paper details our system for the JUST-NLP 2025 Shared Task on English-to-Hindi Legal Machine Translation. We propose a novel two-stage, data-centric approach. First, we annotate the training data by translation difficulty and create easy and hard subsets. We perform SFT on the easier subset to establish a robust "cold start". Then, we apply RLVR exclusively on the harder subset, using machine translation metrics as reward signals. This strategy allowed our system to significantly outperform strong baselines, demonstrating the capability of our systems for machine translation tasks. Source code and model weights are available at https://github.com/ppaolong/FourCorners-JustNLP-MT-Shared-Task

1 Introduction

The Indian legal system presents a compelling machine translation challenge, requiring the translation of complex, jargon-heavy English into accessible Hindi to ensure judicial transparency and access to justice. The standard approach, Supervised Fine-Tuning (SFT), is suboptimal for this task as it tends to overfit by memorizing reference translations and inefficiently treats all training examples equally.

To overcome these challenges, we propose a hybrid SFT-RLVR pipeline guided by a data curriculum. We first employ an external LLM to annotate the training data by translation difficulty. A robust baseline model is then established via SFT on the "easy-to-medium" subset. Finally, we apply Reinforcement Learning with Verifiable Rewards (RLVR) exclusively on the "hard" subset, using the competition's MT evaluation metrics (BLEU, ROUGE, ChrF++) as direct, low-cost reward signals.

Our contributions are threefold:

1. We present a top-performing system for the JUST-NLP 2025 L-MT shared task.
2. We introduce a practical and effective data curriculum strategy that uses an LLM to segment data by difficulty for a hybrid SFT-RLVR training pipeline.
3. We provide empirical evidence that this approach leads to superior performance compared to standard SFT baselines in the specialized legal domain.

† These authors contributed equally as co-first authors.

2 Related Work

2.1 Reinforcement Learning with Verifiable Rewards

While Supervised Fine-Tuning (SFT) is a standard baseline, reinforcement learning (RL) offers a compelling alternative to move beyond the limitations of token-level mimicry and improve model generalization. The traditional RLHF pipeline, often using Proximal Policy Optimization (PPO) (Schulman et al., 2017), is bottlenecked by its reliance on a separately trained value model. This has motivated the development of simpler, value-model-free alternatives like Group Relative Policy Optimization (GRPO) (Shao et al., 2024), which computes advantages by comparing rewards across multiple samples. Subsequent refinements like Dr. GRPO (Liu et al., 2025), GSPO (Zheng et al., 2025), and DAPO (Yu et al., 2025) have further improved the stability and efficiency of this paradigm.

2.2 Reference-Based Metrics as Rewards for Text Generation

These RLVR frameworks make it practical to use automated reference-based metrics directly as reward signals. A significant advance was demonstrated by Chang et al. (2025), who showed that using BLEU (Papineni et al., 2002) as a direct reward for GRPO can match the performance of complex, human-trained reward models for general instruction following. This "metric-as-reward" principle has been effective in specialized domains like legal

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 101–106
December 24, 2025

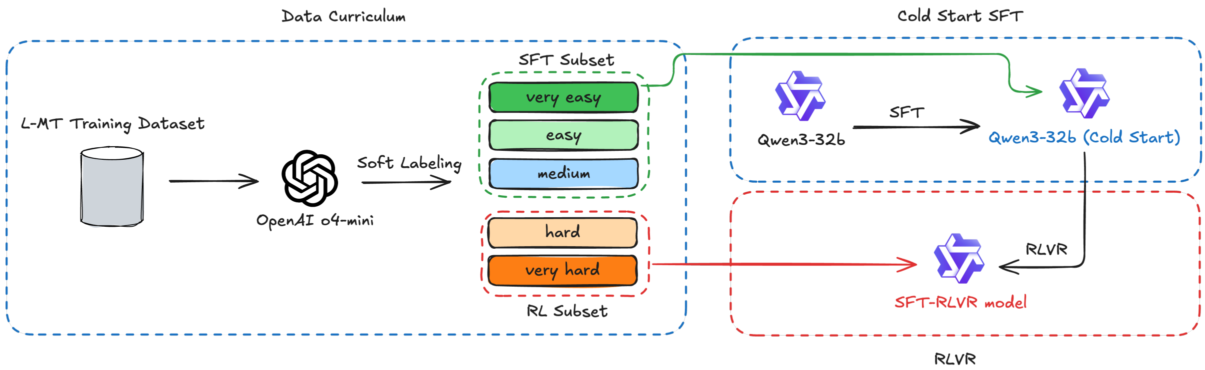


Figure 1: Overview of our proposed system.

question answering (Akarajaradwong et al., 2025). Concurrent to our work, Wang et al. (2025) also employ a two-stage SFT-RL pipeline for literature translation. However, their approach differs significantly, targeting subjective "free translation" and relying on a complex, LLM-as-a-judge (DeepSeek-v3 (DeepSeek-AI et al., 2025)) for its reward signal. Our work is situated within this context, but we apply a data-centric curriculum and use simple, verifiable MT metrics (including ROUGE (Lin, 2004) and ChrF++ (Popović, 2017)) to achieve high-fidelity translation in the precise legal domain.

3 Methodology

Our system employs a two-stage, data-centric pipeline designed to directly optimize a Qwen3-32B model (Yang et al., 2025) for the translation task, as depicted in Figure 1. The process involves meticulous data preparation followed by a hybrid training strategy of Supervised Fine-Tuning (SFT) and Reinforcement Learning with Verifiable Rewards (RLVR).

3.1 Data Preparation and Curriculum

First, we subjected the provided corpus to a comprehensive preprocessing pipeline. This involved normalizing the text by decoding HTML entities, standardizing quotation marks, and replacing ligatures. We then standardized the structure by correcting punctuation spacing, unifying list markers, fixing hyphenation, and collapsing all whitespace.

Inspired by the idea that training should focus on progressively harder examples (Ji et al., 2025), we created a data curriculum. Using o4-mini*, we annotated the 50,000 sentence pairs with five difficulty levels (detailed in Table 1). We then designated the 43,181 "easy-to-medium" pairs as the **SFT Subset** for initial fine-tuning, and the remaining 6,819 "hard" pairs as the **RL Subset** for subsequent reinforcement learning.

Difficulty Level	Count
SFT Subset	
very_low	4167
low	18860
medium	20154
RL Subset	
high	6812
very_high	7

Table 1: Distribution of difficulty labels in the training data.

3.2 Hybrid Training Pipeline

Our training process unfolds in two distinct stages. First, we perform a Cold-Start SFT on the SFT Subset. This efficiently adapts the base model to the legal domain's vocabulary and style using a standard cross-entropy objective, creating a strong foundation. Subsequently, we apply Metric-Driven RLVR exclusively on the RL Subset, using the model from Stage 1 as our policy. In this phase, we update the model to directly maximize rewards derived from standard MT evaluation metrics.

3.3 Reward Function Formulation

The reward signal for the RLVR stage is a weighted sum of primary and auxiliary components.

Primary Metric Rewards: The core of our reward signal was derived from a combination of MT evaluation metrics:

- **BLEU** (Papineni et al., 2002) [0.0-1.0]: Measures n-gram precision for fluency and ade-

*o4-mini-2025-04-16

quacy.

- **Composite ROUGE** (Lin, 2004) [0.0-1.0]: The average F1-score of ROUGE-1, ROUGE-2, and ROUGE-L, providing a comprehensive recall-oriented signal.
- **ChrF++** (Popović, 2017) [0.0-1.0]: A character-level metric robust to morphological variation.[†]

Auxiliary Quality Rewards: To ensure the model produced well-formed outputs, we included two auxiliary rewards:

- **Format Check Reward** [0, 1]: A binary reward that penalizes outputs that do not follow the expected format.
- **Allowed Character Reward** [0, 1]: A binary reward that penalizes the generation of invalid characters in the target Hindi script.

4 Experimental Setup

Statistic	English (Source)	Hindi (Target)
Number of Sentences	50,000	50,000
Total Tokens	1,492,721	1,560,783
Average Sentence Length (Tokens)	29.9	31.2
Median Sentence Length (Tokens)	28.0	29.0
Max Sentence Length (Tokens)	79	71
Type-Token Ratio (TTR)	0.02	0.02

Table 2: Key statistics of the L-MT training dataset.

4.1 Dataset and Evaluation Metrics

All experiments use the official dataset: WMT25 Legal Domain Test Suite (Singh et al., 2025), strictly adhering to the competition's rule of no external data. The training set of 50,000 English-Hindi pairs is characterized by long, complex sentences (avg. 30 tokens) and a highly specialized, repetitive vocabulary (TTR of 0.02), as detailed in Table 2. This dual challenge of syntactic complexity and lexical precision motivated our two-stage approach. Model performance was ranked on the held-out test set using a combined score of the official metrics: BLEU, ROUGE-L, and ChrF++.

4.2 Models

4.2.1 Baselines

We benchmarked our two-stage SFT-RLVR systems against two strong baselines, all using Qwen3-32B (Yang et al., 2025) as the backbone:

- **Base Model:** The pre-trained Qwen3-32B without any fine-tuning.
- **Full SFT:** A strong baseline fine-tuned on the entire 50,000-pair training set.

4.2.2 Our Proposed Systems

Our proposed systems follow the two-stage pipeline described in Section 2. Due to the no-external-data rule, we did not synthesize chain-of-thought and generated translations directly, unlike concurrent work (Wang et al., 2025). We experimented with four primary reward signals for the RLVR phase:

1. **SFT-RLVR-BLEU:** Uses only the BLEU score as a reward signal.
2. **SFT-RLVR-ROUGE:** Uses only the ROUGE composite score as the reward signal.
3. **SFT-RLVR-ChrF++:** Uses only the ChrF++ score as the reward signal.
4. **SFT-RLVR-Combined:** Uses an equally weighted average of BLEU, ROUGE, and ChrF++ scores as the primary reward signal.

Auxiliary Quality Rewards (from Section 3.3) were consistently applied on our proposed systems to ensure the generation of well-formed outputs.

Parameter	SFT	RLVR
Base Model	Qwen3-32b	Qwen3-32b (cold start)
Quantization	4-bit	4-bit
Optimizer	AdamW	AdamW
Learning Rate	1e-4	1e-4
Batch Size	32	1
Rollout	-	32
Numbers of Epochs	3	1
Scheduler	cosine	cosine
LoRA Rank	256	1
LoRA α	256	1
Loss Type	cross-entropy	DAPO

Table 3: Key hyperparameters for the Supervised Fine-Tuning (SFT) and Reinforcement Learning with Verifiable Rewards (RLVR) stages. Other parameters use its default value from trl (von Werra et al., 2020).

4.3 Implementation Details

We implemented our pipeline using the unsloth library (Daniel Han and team, 2023) for efficient training, with stanza (Qi et al., 2020) for tokenization and trl's (von Werra et al., 2020) implementation of the DAPO algorithm (Yu et al., 2025) for the RL phase. All models were fine-tuned using LoRA (Hu et al., 2021), with key hyperparameters detailed in Table 3. All models were trained on A100 80GB GPU.

[†]The standard ChrF++ score (0-100) is normalized to a [0, 1] range.

Model	BLEU	ROUGE-L	ChrF++	Joint Score
Qwen3-32b (baseline)	0.114	0.346	0.405	0.865
Qwen3-32b Full SFT (strong baseline)	0.446	0.624	0.708	1.778
Qwen3-32b Cold-Start SFT	0.440	0.618	0.702	1.760
+RLVR-BLEU (best system)	**0.501**	**0.657**	**0.742**	**1.900**
+RLVR-ROUGE	0.475	0.639	0.720	1.834
+RLVR-ChrF++	0.492	0.654	**0.742**	1.888
+RLVR-Combined	0.495	0.654	0.741	1.890

Table 4: Performance of all models on the test set from codabench submission system. Joint Score is the sum of BLEU, ROUGE-L, and Chrf++ scores. Our best system is highlighted in bold.

For the RLVR stage, we set the LoRA rank and α to 1. This choice is informed by Schulman and Lab (2025) findings that the sparse, per-sequence reward signal from policy gradient methods requires significantly less adapter capacity than the dense, per-token signal of SFT .

5 Results and Discussion

5.1 Overall Performance

The results, presented in Table 4, confirm the superiority of our two-stage SFT-RLVR pipeline (see final official leaderboard in Table 5). All RLVR variants significantly outperform the Full SFT baseline, validating our hybrid approach. Our top-performing system, SFT-RLVR-BLEU, demonstrating that a data-centric curriculum followed by direct metric optimization is a highly effective strategy.

Precision-Oriented Rewards Excel: The primary factor differentiating the performance of our RLVR models was the choice of reward signal. We find a clear advantage for precision-oriented rewards. The top results were achieved by models trained on BLEU and ChrF++, which directly penalize deviations in specific terminology and structure, a critical requirement for maintaining fidelity in the legal domain.

Recall-Oriented Rewards Are Less Effective: On the other hand, the model rewarded with the ROUGE score was our least effective. While this model still beat the baseline, its focus on the overall "gist" of the translation is a poor fit for legal text, where exact wording is critical.

Combining Rewards May Be Suboptimal: Interestingly, using only the BLEU score as a reward was also more effective than combining all three metrics. This suggests that giving the model a single, clear goal for precision works better than a mixed signal.

Our key takeaway is that for a high-stakes field like law, the winning strategy is to directly reward the model for getting the exact words right. We further observe the relative stability of each reward during RLVR phase in Appendix A.

5.2 Ablation Study

Furthermore, an ablation study validates the efficiency of our two-stage design. In Table 4, SFT on Easy Data model alone performs competitively with, though slightly below, the Full SFT baseline. This demonstrates that our "Cold-Start" SFT phase effectively creates a strong foundation. The subsequent targeted RLVR phase not only recovers this minor deficit but elevates the model's performance considerably across all metrics.

6 Conclusion

This paper presents a top-performing system for the JUST-NLP 2025 English-to-Hindi Legal MT shared task. Our approach overcomes the limitations of standard Supervised Fine-Tuning with a two-stage, data-centric pipeline: a Cold-Start SFT on an automatically curated easy subset, followed by Reinforcement Learning with Verifiable Rewards (RLVR) on the harder subset. By directly optimizing for MT metrics as rewards, notably the precision-oriented BLEU, our system significantly outperformed a strong SFT baseline.

Our work validates direct metric optimization, guided by a data curriculum, as a powerful and efficient strategy for developing state-of-the-art systems in specialized domains. Future directions include exploring adaptive reward schemes and applying this methodology to other high-stakes NLP tasks.

Limitation

Our approach has several limitations. First, due to computational and time constraints, we were

Rank Possition	Team Name	Affiliation	Country	BLEU↑	METEOR↑	TER↓	CHRF++↑	BERTScore↑	COMET↑	AutoRank↑	Leaderbord Name
1	Team-SVNIT	Sardar Vallabhbhai National Institute of Technology, Surat	India	51.61	75.8	37.09	73.29	92.61	76.36	61.62	rupeshdhakad06
2	FourCorners	VISAI AI	Thailand	50.19	69.54	42.32	73.67	92.7	75.74	60.31	pawitsapak
3	goodmen	Sardar Vallabhbhai National Institute of Technology, Surat	India	48.56	67.15	41.63	73.07	92.38	75.16	59.39	skdrj123
4	JUNLP	Jadavpur University	India	46.03	71.84	42.08	70.59	91.19	73.72	58.90	iamamit
5	JUST-MEI	SOA University	India	46.67	72.86	44.63	70.03	90.86	72.12	58.79	lsmeetei
6	Lawgorithms	Thangal Kunju Musaliyar College of Engineering	India	46.27	71.8	43.06	68.32	91.03	72.14	58.26	sreehari_saji
7	Tokenizers	Sardar Vallabhbhai National Institute of Technology, Surat	India	34.08	61.78	55.25	56.75	87.39	65.2	50.87	tokenizers

Table 5: Official Leaderboard of JustNLP MT Shared Task (https://exploration-lab.github.io/JUST-NLP/JustNLP25_L-MT_Result.pdf).

unable to explore the potential of other metrics to use as a reward for RLVR such as BERTScore. Our reported results are based on a limited exploration of model hyperparameters and data combinations. Second, our data curriculum's reliance on a proprietary LLM for difficulty annotation impacts the full reproducibility and transparency of our pipeline. Finally, the competition's evaluation protocol, which provided only aggregate scores via the CodaBench platform, precluded a qualitative, example-by-example analysis of our model's improvements over the baselines.

References

Pawitsapak Akarajaradwong, Chompakorn Chaksangchaichot, Pirat Pothavorn, Ekapol Chuangsuwanich, Attapol Rutherford, and Sarana Nutanong. 2025. Aligning LLMs for Thai legal question answering with efficient semantic-similarity rewards. In *Proceedings of the Natural Legal Language Processing Workshop 2025*, pages 304–316, Suzhou, China. Association for Computational Linguistics.

Yapei Chang, Yekyung Kim, Michael Krumdick, Amir Zadeh, Chuan Li, Chris Tanner, and Mohit Iyyer. 2025. Bleuberi: Bleu is a surprisingly effective reward for instruction following. *Preprint*, arXiv:2505.11080.

Michael Han Daniel Han and Unsloth team. 2023. Unsloth.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, and 181 others. 2025. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Yunjie Ji, Sitong Zhao, Xiaoyu Tian, Haotian Wang, Shuaiting Chen, Yiping Peng, Han Zhao, and Xiangang Li. 2025. How difficulty-aware staged reinforcement learning enhances llms' reasoning capabilities: A preliminary experimental study. *Preprint*, arXiv:2504.00829.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. 2025. Understanding r1-zero-like training: A critical perspective. *Preprint*, arXiv:2503.20783.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Maja Popović. 2017. chrF++: words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation*, pages 612–618, Copenhagen, Denmark. Association for Computational Linguistics.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, Online. Association for Computational Linguistics.

John Schulman and Thinking Machines Lab. 2025. Lora without regret. *Thinking Machines Lab: Connectionism*. Https://thinkingmachines.ai/blog/lora/.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *Preprint*, arXiv:1707.06347.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *Preprint*, arXiv:2402.03300.

Kshetrimayum Boynao Singh, Deepak Kumar, and Asif Ekbal. 2025. Evaluation of llm for english to hindi legal domain machine translation systems. In *Proceedings of the Tenth Conference on Machine Translation (WMT 2025)*, pages 823–833, Suzhou, China. Association for Computational Linguistics.

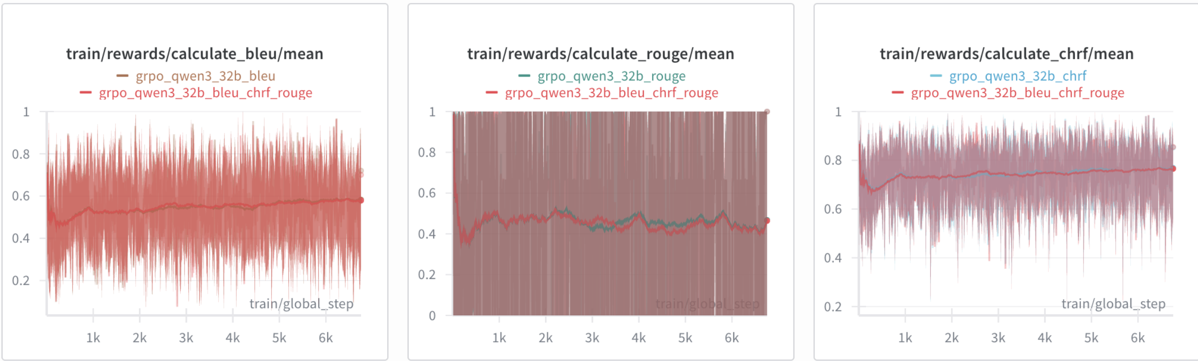


Figure 2: Mean reward signals during the RLVR phase for models trained with single-metric and combined rewards. (Left) The BLEU reward shows a stable, gradual increase. (Center) The composite ROUGE reward is highly erratic and unstable, with no clear upward trajectory. (Right) The ChrF++ reward, similar to BLEU, exhibits a strong and consistent increasing trend. The x-axis represents the training steps, and the y-axis represents the normalized reward score.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. 2020. Trl: Transformer reinforcement learning. `https://github.com/huggingface/trl`.

Jiaan Wang, Fandong Meng, and Jie Zhou. 2025. Deep-trans: Deep reasoning translation via reinforcement learning. *Preprint*, arXiv:2504.10187.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, and 16 others. 2025. Dapo: An open-source llm reinforcement learning system at scale. *Preprint*, arXiv:2503.14476.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. 2025. Group sequence policy optimization. *Preprint*, arXiv:2507.18071.

A Analysis of Reward Signal Stability During RLVR

To provide further insight into the training dynamics of our RLVR models, we plot the mean reward signals for BLEU, our composite ROUGE score, and ChrF++ over the course of training. Figure 2 illustrates the stability and progression of these metrics as rewards.

A key observation from these learning curves is the difference in the stability of the reward signals. As seen in the left and right panels of Figure 2, the mean rewards for BLEU and ChrF++ exhibit a clear and stable upward trend throughout the training process. Although individual batch rewards are noisy (indicated by the wide, faint bands), the smoothed average consistently improves, demonstrating that the model is successfully learning a policy that optimizes for these precision-oriented metrics.

In contrast, the middle panel shows that the composite ROUGE reward is highly unstable. The learning curve is erratic and jagged, with no sustained upward trajectory. This instability suggests that the optimization landscape for a recall-oriented metric like ROUGE is less smooth for this task. The model struggles to find a consistent policy that reliably increases the ROUGE score, possibly because the reward signal is less sensitive to the incremental, precision-focused improvements that are easier for the model to learn.

This empirical observation further supports our main finding in Section 5: that precision-oriented metrics like BLEU and ChrF++ not only lead to better final evaluation scores but also provide a more stable and effective training signal for the RL agent in the high-fidelity legal translation domain.

Contextors at L-SUMM: Retriever-Driven Multi-Generator Summarization

Pavithra Neelamegam, S. Jaya Nirmala
Department of Computer Science and Engineering
National Institute of Technology, Tiruchirappalli, Tamil Nadu, India
{406424001, sjaya}@nitt.edu

Abstract

Indian court judgments are very difficult to automatically summarize because of their length, complex legal reasoning and scattered important information. This paper outlines the methodology used for the Legal Summarization (L-SUMM) shared task at the JUST-NLP 2025 Workshop, which aims to provide abstractive summaries of roughly 500 words from English-language Indian court rulings that are logical, concise and factually accurate. The paper proposes a Retriever-Driven Multi-Generator Summarization framework that combines a semantic retriever with fine-tuned encoder–decoder models BART, Pegasus and LED to enhance legal document summarization. This pipeline uses cosine similarity analysis to improve summary faithfulness, cross-model validation to guarantee factual consistency and iterative retrieval expansion to choose relevant text chunks in order to address document length and reduce hallucinations. Despite being limited to 400–500 words, the generated summaries successfully convey legal reasoning. Our team *Contextors* achieved an average score of 22.51, ranking 4th out of 9 in the L-SUMM shared task leaderboard, demonstrating the efficacy of Retriever-Driven Multi-Generator Summarization approach, which improves transparency, accessibility, and effective understanding of legal documents. This method shows excellent content coverage and coherence when assessed using ROUGE-2, ROUGE-L, and BLEU criteria.

1 Introduction

Finding an important information in lengthy, complex and unstructured court case judgments can be difficult for legal professionals. These documents are often hundreds of words long and need a lot of time and effort to read and understand. Such documents need to be manually summarized, and it is a costlier and time-consuming process. It also requires expert legal knowledge. To address the challenge of processing lengthy legal judgments, the JUST-NLP 2025 Workshop conducted the Shared Task on Legal Summarization (L-SUMM) .

This work promotes the creation of AI-powered solutions that can automatically produce abstractive summaries of Indian court rulings. Abstractive summarizing creates new, coherent text that translates, condenses and rephrases complicated legal jargon into instructive summaries of about 500 words that capture the core of the ruling, in contrast to extractive summary, which chooses preexisting sentences from a document.

This approach uses transformer-based encoder–decoder designs like BART, T5, Pegasus and LED, which are further enhanced by retriever models based on cosine similarity analysis for semantic chunk selection. These models capture the unique terminology, reasoning processes and discourse structures of Indian court decisions by fine-tuning on domain-specific legal data.

Submissions to the shared work are evaluated using standard relevance metrics, such as ROUGE-2, ROUGE-L and BLEU, which evaluate the quality, fluency and overlap of the generated summaries with human references.

2 Related Work

For Indian legal papers, (Ghosh et al., 2022) suggested a text normalization method that standarizes reference styles, legal jargon and acronyms before fine tuning generic models. But, it fails to capture long-document connections and hierarchical structures. (Deroy et al., 2024) examined the LLMs for summarizing judgments by contrasting extractive and abstractive approaches but suffered from factual errors and limited citations awareness. (Santosh et al., 2024) presented LexAbSumm, an aspect-based framework improving interpretability, but has a trouble handling inter aspect interdependence.

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 107–112
December 24, 2025

BART (Lewis et al., 2019) achieves strong summarization quality but is constrained by its limited 1K token input window. Pegasus (Zhang et al., 2020) better aligns with summarization tasks. However, its context window and domain generality limit applicability to legal reasoning tasks. Furthermore, after tuning Pegasus on domain-specific corpora, Legal Pegasus(Sharma and Singh, 2024) enhanced factual consistency but failed to sustain complex citation relations.To address long-document contexts, BigBird (Zaheer et al., 2020) proposed a sparse-attention mechanism for efficient scaling and LED (Beltagy et al., 2020) extended this idea to handle up to 16K tokens, enhancing length coverage, but both the approaches struggle in capturing hierarchical legal semantics.

Several transformer-based summarizers (BART, T5, Pegasus) were merged in ensemble frameworks (Albayati et al., 2025) to improve factuality. However, it provides only a minimal improvement at the expense of significant computational complexity. More recently, RAG models (Ajay Mukund and Easwarakumar, 2025) have integrated external knowledge retrieval with generative summarization, however, still face issues related to retrieval precision, latency, and maintaining structural coherence in lengthy legal texts. Further research is required to capture the facts, issues, laws and other crucial components.

Our approach differs from previous legal summarization methods by introducing a retrieval-guided, multi-generator framework. It first uses InLegalBERT to select the most relevant judgment segments, then combines outputs from multiple fine-tuned abstractive models instead of relying on a single generator. A faithfulness-based semantic alignment score is finally used to choose the most accurate summary, resulting in a retrieval-aware and fact-faithful summarization method not present in prior work.

3 Dataset Description

The InLSum (Indian Legal Summarization) dataset, used for this shared task, includes 1200 training samples, 200 validation samples and 400 test samples of Indian court rulings that are accompanied with abstractive summaries produced by legal experts. The InLSum dataset is provided in JSONL format.The training dataset contains judgment and reference summary files. The validation and test dataset contains only judgments.

4 Model Description

4.1 Fine-Tuned Pre-trained Models

A variety of pre-trained sequence-to-sequence models, such as BART, Legal-Pegasus, T5 and LED, has been finetuned using InLSumm dataset. Every judgment and summary is preprocessed to remove extraneous text, page numbers and repeating blocks, so that the models can focus on the most important information. During the training phase, 90% of dataset is used for training and 10% of dataset is used for validation purpose. The models are adjusted to produce an abstract summary of 400–500 words. ROUGE-2, ROUGE-L and BLEU are used to evaluate overall quality of the summarization.

4.2 Legal Ensemble Summarization Framework

This work presents an ensemble based abstractive summarization framework for legal judgments that integrates multiple fine tuned transformer architectures. Each judgment is preprocessed to remove unnecessary components such as case identifiers, citations, and formatting patterns. In the ensemble configuration, each model generates a summary for the same input independently. BART and Pegasus, BART and LED, and Pegasus and LED are pairwise ensembled to take advantage of the strengths of BART's fluency, LED's sparse attention for lengthy sequences, and Pegasus's summary pre-training. Clarity, sentence versatility and legal relevance are further improved by a 3-way hybrid ensemble that combines BART, Pegasus, and LED.

Semantic ranking is used for selection using InLegalBERT(Sharma and Singh, 2024). Sentence embeddings will be calculated for both candidate summaries and the initial judgment. The final result is then determined by selecting the summary with the highest cosine similarity. In order to offer a thorough assessment of summarization quality, the final evaluation employs ROUGE-2, ROUGE-L, and BLEU, which examine text fluency, structure similarity, and content accuracy.

4.3 Retriever-Driven Multi-Generator Summarization

The proposed Retriever-Driven Multi-Generator Summarization framework, offers a novel method for producing concise, logical, and factually consistent summaries of complicated legal rulings. Semantic retrieval, multi-generator fine-tuning, and

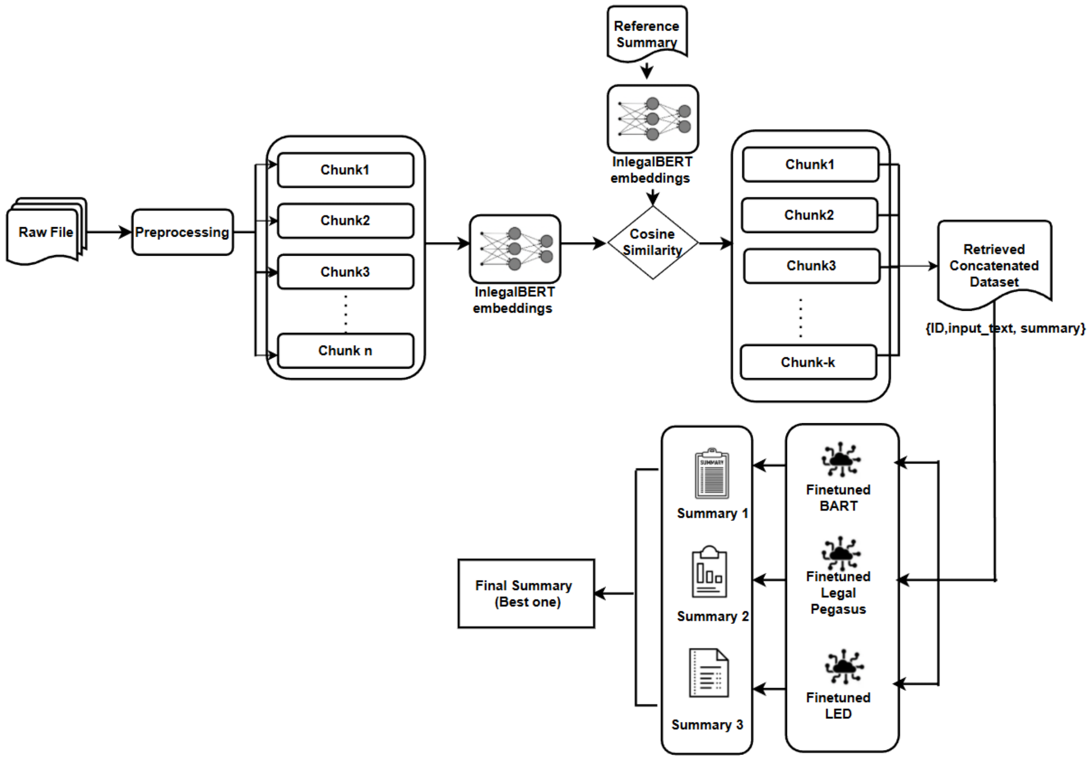


Figure 1: Retriever-Driven Multi-Generator Summarization framework

automatic ensemble selection are the three main components of this framework.As shown in Figure 1, this proposed framework combines retrieval and generation modules.

After preprocessing, to maintain compatibility with transformer input restrictions, each file is split into reasonable chunks (chunk$_1$, chunk$_2$,..., chunk$_n$) due to the length of legal documents. Judgments are split into 8-sentence chunks, embedded using 768-D vectors, and ranked by cosine similarity against the full-judgment embedding to measure semantic relevance. The system retrieves the top 8 most relevant chunks for summary generation. The obtained dataset is composed of tuples of ID, input_text, summary. By ensuring that only parts of the document that are most contextually aligned contribute to model fine-tuning, this retrieval procedure improves efficiency and factual grounding.

90% of training data and 10% of validation data are created from the retrieved and concatenated dataset. This dataset is used to independently train three transformer-based summarization models LED, Legal Pegasus and BART.

Each finetuned model independently produces a summary for the same input document in the ensemble setting. The embeddings obtained from the validation judgment datatsets are then compared to these outputs (Summary1, Summary2, and Summary3) using cosine similarity. In order to ensure contextual accuracy and relevance, the summary with the best semantic similarity score is chosen as the final summary. The chosen final summaries are statistically assessed for n-gram overlap, fluency, and informativeness using the average of the ROUGE-2, ROUGE-L, BLEU scores.

Training Phase: To find the most important textual parts, the system uses chunk–summary similarity in a supervised retrieval-enhanced training configuration. From these retrieved pieces, the summarizing model is subsequently refined to provide summaries.

Inference Phase: Since gold summaries are not available, the retrieval process runs unsupervised during inference. In order to extract the most semantically representative chunks, the model calculates the mean embedding of the input judgment. These chunks are then supplied to the refined summarizer to provide abstractive summaries.

All preprocessing scripts, codes used in this work are publicly available online. [1]

[1] https://github.com/pavithraneelamegam/Summarization

Table 1: Performance comparison of fine-tuned and ensemble models using ROUGE and BLEU metrics.

Model	ROUGE-2	ROUGE-L	BLEU	AVG
Fine-Tuned Models				
Fine-Tuned BART	17.79	25.32	14.53	19.21
Fine-Tuned Legal Pegasus	17.64	25.12	13.66	18.81
Fine-Tuned LED	16.74	24.02	12.51	17.76
Fine-Tuned T5	16.08	23.02	11.03	16.71
Ensemble Models				
Fine-Tuned (BART and Pegasus)	23.02	25.20	15.47	21.23
Fine-Tuned (BART and LED)	23.40	25.40	16.06	21.62
Fine-Tuned (BART, Pegasus and LED)	23.62	25.60	16.69	21.97
Retriever-Driven Multi-Generator Summarization	**25.13**	**25.59**	**16.8**	**22.51**

5 Evaluation Metrics

5.1 ROUGE Scores

Automatic summarization evaluation is the main application of the ROUGE (Recall-Oriented Understudy for Gisting Evaluation) metric. By computing n-gram recall, it determines how much the generated result summaries and reference summaries coincide. Equation 1 represents ROUGE-1 (unigram).

$$\mathrm{ROUGE}_1 = \frac{\sum_{\text{unigram}\in\text{reference}} \text{Count}_{\text{match}}(\text{unigram})}{\sum_{\text{unigram}\in\text{reference}} \text{Count}(\text{unigram})} \tag{1}$$

5.2 BLEU Score

BLEU (Bilingual Evaluation Understudy) captures sufficiency, fidelity, and fluency by measuring the amount of overlap of n-grams among a reference phrase and a candidate (hypothesis).

6 Results and Discussion

Table 1 presents the comparative performance of individual and ensemble summarization models on the InLSum dataset. BART outperformed the other finetuned models (AVG = 19.21), closely followed by Legal Pegasus (AVG = 18.81). This indicates that both architectures effectively model hierarchical dependencies and domain-specific semantics in legal documents. LED showed slightly lower performance (AVG = 17.76), likely due to the truncation of lengthy case texts and difficulty in handling factual segments.

The BART and LED combination showed a notable improvement (AVG = 21.62) because LED's contextual comprehension supports BART's improved surface realization and syntax fluency. The triple-model ensembles (BART, Pegasus and LED) produced the highest average (AVG = 21.97) , suggesting that integrating different attention and producing mechanisms results in more balanced summaries with enhanced coverage and factual coherence.

Outperforming all other models, the proposed Retriever-Driven Multi-Generator Summarization framework produced the best overall results (ROUGE-L = 25.59, BLEU = 16.8, AVG = 22.51). Each generator can now concentrate on the most semantically relevant context pieces prior to generation due to the retrieval-enhanced architecture. It generates summaries that are context-aware and factually consistent, better capturing interdependent facts and fine-grained legal reasoning than single models.

6.1 Qualitative Analysis

The generated summaries generally capture the main storyline of judicial decisions, including the key dispute, parties, and major facts. This shows that the model handles high-level narrative extraction well. However, it often misses important legal details such as statutes, procedural steps, and the Court's reasoning and may overlook subtle facts or introduce errors. As a result, the summaries are coherent but not fully accurate from a legal perspective.

These challenges show why abstractive summarization in law is difficult. Judgments are complex, detailed, and heavily structured around legal reasoning. Automated systems often struggle to preserve the logical flow, handle layered arguments, or capture references to past cases. Therefore, even though the summaries offer a good general narrative, they still require human review and legal expertise to ensure accuracy and reliability.

In the first example, as shown in Figure 2 the model captures the main theme well, correctly summarizing the dispute over conducting exams with COVID-19 safety measures. However, it misses key legal details such as the statutory basis, opt-out rules, and specific safety guidelines making the summary less accurate and complete from a legal perspective.

ORIGINAL JUDGMENT
{"id": "id_1011", "judgment": ..... For the benefit of around 3.74 lakh\ncandidates, the ICAI is trying to ensure that the examinations are h eld in July, 2021\nwith all mandated Covid -19 safety protocols in place.\These are neither possible nor practical.\n• The Institute has issued Guidelines for\nExamination Centres, Examination\nfunctionaries and Candidates (Annexure\n'B'), which are very comprehensive and in\nfull compliance with Gover nment of India\nguidelines, with focus on social distancing,\nthermal scanning and adequate facilities of\nface mask, sanitizer, etc.\n• .... There is no reason\nto believe that a responsible Institute like the ICAI would not take all possible safety\nprecautions for c onduct of the examinations...Guidelines for Candidates GENERAL GUIDELINES\nICAI is taking all possible measures for safe and secure conduct of CA\nExaminations sched uled to be held in J uly,2021 . ICAI has increased the\nexaminations centres to cover additional 192 districts of the country to\nfacilitate candidates to not to travel beyond their district to the extent possible.....\}
GENERATED SUMMARY
{"ID": "id_1011", "Summary": ....The ICAI has also issued Guidelines for Examination Centres, Examination functionaries and Candidates for the July 2021 CA Examinations, which are in conformity with the Government of India guidelines, with focus on social distancing, thermal scanning and adequate facilities of face mask, sanitizer and water bottles to be provided to the candidates in case of need, the Court said. .... It was submitted that the exam centres had increased the capacity of examination centres to cover more 192 districts in the country, and that it was in the interest of around 3.74 lakh candidates to be able to participate in the examinations..... }

Figure 2: original judgment text vs generated summary(sample 1)

In the second example as shown in Figure 3, the model captures the main narrative and human aspects of the judgment well, but it misses key legal reasoning, overlooks the age-related dispute, and includes minor inaccuracies. As a result, the summary conveys the story but lacks the legal precision needed to fully reflect the Court's reasoning.

ORIGINAL JUDGMENT
{"id": "id_1074", "judgment": ....who\nwas allegedly a minor as per the State and Mr. Arif Khan, who eloped\nat the calling of their love . Oblivious of the demands of law, they got\nmarried as per Muslim rites and ceremonies, since they both belonged\nto the same religion, with the blessi ngs of parents of Mr. Khan as the\nparents presumed that the marriage between the parties co uld be\nsolemnized and respected the feeling of love and affection between Ms.\n'A' and their son. ...the entry o f investigating agency, which was to work as per existing\nlaw. At the time of recovery of Ms. 'A' , she was five months pregnant .\nShe refused to abort the child as it was born out of her marital u nion\nand love for Mr. Khan , and in her statements before poli ce as well as\nbefore the Magistrate , she stood by her stand of love for the man, she\nhad married.....}
GENERATED SUMMARY
{"ID": "id_1074", "Summary": "... Justice Subramonium Prasad noted that the woman was five months pregnant at the time of arrest and refused to abort the child as it was born out of her love for the man, and in her statements before the police and the Magistrate, she stood by her stand that she had married the man. "......the Court said. The woman and her husband had eloped at the calling of their love and got married as per Muslim rites and ceremonies, oblivious of the demands of law, since they both belonged to the same religion, with the blessing of the parents of the man as the parents presumed that the marriage between the parties would be solemnized and respected the feeling of love and affection between the woman and their son. ....}

Figure 3: original judgment text vs generated summary(sample 2)

7 Conclusion

The JustNLP Shared Task underscores the potential of NLP techniques in addressing the challenges of abstractive summarization for complex legal documents. In this study, the proposed Retriever-Driven Multi-Generator Summarization framework that integrates multiple fine-tuned transformer models with a retrieval based preprocessing pipeline to generate coherent, legally faithful and semantically rich summaries. This ensemble architecture outperforms individual fine-tuned models, yielding an overall improvement of +2-3 ROUGE points over evaluation criteria. To further enhance summary reliability and interpretability, future research will concentrate on hierarchical retrieval, fidelity optimization based on reinforcement learning, and legal provision-aware structural modeling.

Limitations

There is a trade-off between faithfulness and abstraction in the suggested Retriever-Driven Multi-Generator Summarization framework. Accuracy at the sentence level cannot be guaranteed by existing embedding-based checks. The quality of retrieval has a significant impact on the proposed framework's performance as well. The model treats judgments as plain text rather than acknowledging their hierarchical structure, rhetorical functions and argument flow, all of which are crucial for creating logical and legally accurate summaries. Lastly, although retrieval introduces only moderate overhead, the use of multiple large transformer models makes the overall pipeline computationally intensive. A precise complexity analysis is difficult due to architecture-dependent embedding and generation costs. Therefore, we plan to include detailed profiling and computational analysis in future work.

Acknowledgements

This work was funded by the Science and Engineering Research Board (SERB), Department of Science and Technology, Government of India, under Grant No. CRG/2023/007683.

References

S Ajay Mukund and KS Easwarakumar. 2025. Optimizing legal text summarization through dynamic retrieval-augmented generation and domain-specific adaptation. *Symmetry*, 17(5):633.

Maha Ahmed Abdullah Albayati, Kürşat Mustafa Karaoğlan, and Oğuz Findik. 2025. Towards efficient multi-legal document summarization: An ensemble approach for turkish law. *Engineering Science and Technology, an International Journal*, 70:102138.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Aniket Deroy, Kripabandhu Ghosh, and Saptarshi Ghosh. 2024. Applicability of large language models and generative models for legal case judgement summarization.

Satyajit Ghosh, Mousumi Dutta, and Tanaya Das. 2022. Indian legal text summarization: A text normalization-based approach. In *2022 IEEE 19th India Council International Conference (INDICON)*, pages 1–4.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

T. Y. S. S Santosh, Mahmoud Aly, and Matthias Grabmair. 2024. Lexabsumm: Aspect-based summarization of legal decisions. *Preprint*, arXiv:2404.00594.

Saloni Sharma and Piyush Pratap Singh. 2024. Domain-specific summarization: Optimizing inlegalbert for indian judgment reports.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and 1 others. 2020. Big bird: Transformers for longer sequences. *Advances in neural information processing systems*, 33:17283–17297.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International conference on machine learning*, pages 11328–11339. PMLR.

A Budget Recipe for Finetuning a Long-form Legal Summarization Model

Chompakorn Chaksangchaichot*
VISAI AI, Thailand
chompakornc_pro@vistec.ac.th

Pawitsapak Akarajaradwong*
VISAI AI, Thailand
pawitsapaka_visai@vistec.ac.th

Abstract

We describe an inexpensive system that ranked first in the JUST-NLP 2025 L-SUMM task, summarizing very long Indian court judgments (up to 857k characters) using a single 80GB GPU and a total budget of about $50. Our pipeline first filters out length–summary outliers, then applies two-stage LoRA SFT on Qwen3-4B-Instruct-2507 to learn style and extend context, and finally runs RLVR tuned to BLEU, ROUGE-2, and ROUGE-L, with BLEU upweighted. We showed that two-stage SFT is better than a single-stage run, and RLVR gives the largest gains, reaching 32.71 internal vs. 16.15 base and 29.91 on the test leaderboard. In ablation on prompting, we find that a simple, naive prompt converges faster but saturates earlier, while the curated legal-structured prompt keeps improving with longer training and yields higher final scores, and the finetuned model remains fairly robust to unseen prompts. Our code[1] and models[2] are fully open-sourced, available for reproducibility.

1 Introduction

Summarization is one of the challenging tasks in the Natural Language Processing (NLP) domain (Zhang et al., 2024). Several benchmarks tried to measure different NLP systems using standard metrics n-gram-based metrics (Lin, 2004; Papineni et al., 2002) or LLM-as-judges (Li et al., 2024). In some specific case, summarization can be viewed as a domain-specific problem. For example, medical note summarization (Michalopoulos et al., 2022) in medical domains, meeting summary which usually summarizes key points and agenda (Kirstein et al., 2024), or in legal where summarization was done in legal cases (Datta et al., 2023; Shukla et al., 2022) or court judgments (Sharma et al., 2023). In this work, we focus specifically on legal judgment summarization, which is part of the JUST-NLP 2025 Legal Summarization (L-SUMM) Shared Task. We framed L-SUMM as a long-form summarization task where the length of the judgment can be extremely long (up to 857,477 characters).

To tackle this challenge, we utilized a standard finetuning technique followed by Reinforcement Learning with Verifiable Reward (RLVR). Additionally, we also constraint our compute budget to only $50 in GPU hours and only require a single GPU. To achieve such training efficiency, we utilized supervised finetuning using LoRA (Hu et al., 2021) on Qwen3-4B-Instruct-2507 (Yang et al., 2025). Our pipeline consists of three steps, two-stage supervised finetuning followed by a final RLVR for aligning LLM towards better summarization style. Our results showed that a two-stage supervised finetuning is necessary given a constrained compute budget compared to unified single stage. Furthermore, RLVR significantly boosted the finetuned model performance across all summarization metrics. However, we also noticed the instability in applying RLVR towards legal summarization across long-form summary. In our ablation study, we also investigate the effect of prompting towards downstream performance during supervised finetuning stage. In summary, we conclude our contributions as follows:

1. We provides a detailed system description of our design that ranked first place in the L-SUMM Shared Task.
2. We demonstrate an efficient training pipeline that achieves competitive summarization performance under a strict compute budget roughtly $50 using a single 80GB GPU.
3. We study the effect of prompting supervised finetuning data on legal summarization and report findings.

*These authors contributed equally as co-first authors.
[1] https://github.com/tann9949/justnlp-2025-legal-summ
[2] https://huggingface.co/VISAI-AI/Qwen3-4B-Instruct-2507-L-SUMM-fourcorners-rl-r2-ckpt500

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 113–120
December 24, 2025 ©2025 Association for Computational Linguistics

2 Related Works

Reinforcement Learning with Verifiable Reward Reinforcement Learning with Verifiable Rewards (RLVR) was introduced as a more scalable alternative to the classical RLHF setup (Ouyang et al., 2022), which typically relies on PPO (Schulman et al., 2017), a critic head, and a learned reward model components that make RLHF costly to train. To reduce this overhead, Direct Preference Optimization (DPO) (Rafailov et al., 2024) became popular, but it still requires human preference pairs. Group Relative Policy Optimization (GRPO) (Shao et al., 2024) addressed this by replacing preference data with rule-based, verifiable rewards, thereby removing the need for both a critic head and curated preference pairs. Subsequent works such as Dr. GRPO (Liu et al., 2025b) and DAPO (Yu et al., 2025) improved GRPO's sampling efficiency, reduced length bias, and refined clipping strategies. Group Sequence Policy Optimization (GSPO) (Zheng et al., 2025) further stabilized training for MoE models, by performing advantage renormalization at the sequence level instead of per token.

In this work, we adopt this RLVR to further align an instruction-tuned model for legal summarization, using verifiable task rewards. Prior studies have also shown that n-gram–based rewards (Chang et al., 2025) and semantic-similarity rewards (Akarajaradwong et al., 2025) can be effectively plugged into this framework to strengthen instruction following capabilities.

Parameter Efficient Approaches for Long Contextual Task Recent works proposed an efficient approach to conduct Parameter Efficient Finetuning (PEFT) towards long-form contents. LongLoRA (Chen et al., 2024) proposed using shifted-sparse attention to expand context length cheaply as well as it's quantized counterparts LongQLoRA (Yang, 2023). RST-LoRA (Liu and Demberg, 2024) adapts LoRA for long-document summarization by injecting RST discourse signals (centrality, relations, and their confidence) into the adapter, giving the model a richer, structure-aware fine-tuning signal and outperforming vanilla LoRA and full fine-tuning on multiple evaluations.

Legal Summarization There are several works introduced dataset for legal summarization task. Shukla et al. (2022) released three new case law summarization datasets drawn from Indian and UK Supreme Court judgments, and conducted extensive evaluations of extractive vs abstractive summarization methods on these legal cases. Building on this, Datta et al. (2023) presented the MILDSum corpus, a benchmark of 3,122 Indian legal judgments with high-quality summaries in both English and Hindi. Joshi et al. (2024) introduced IL-TUR, a broad benchmark for Indian Legal Text Understanding that includes a summarization task among eight diverse legal NLP tasks. IL-TUR provides multilingual legal datasets (spanning English and 9 Indian languages) and reports baseline results for each task.

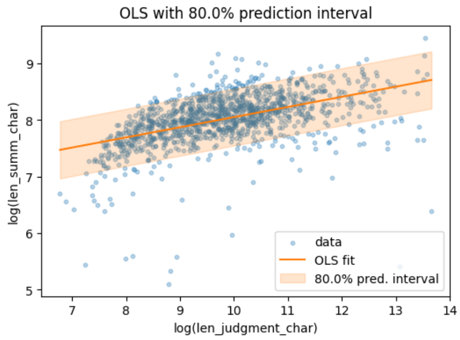


Figure 1: Line best fit and 80% prediction interval between judgment word counts and summary word counts in log-scale.

3 Methodology & Experimental Setups

Our system includes a data preparation phase that removes potential data outliers, followed by a two-stage instruction tuning pipeline before applying a final alignment tuning via RLVR.

3.1 Data Preparation

Upon inspecting the dataset, we found that the ratio between the judgment length and its summarized reference was sometimes disproportionate. In particular, some summaries were longer than the judgment, while others were shorter than expected.

To reduce noise before finetuning, we removed samples whose summary–judgment length ratio deviated from the expected trend. We first fitted a linear regression between judgment word counts and summary word counts in log-scale, then discarded any sample falling outside the 80% prediction interval, as illustrated in Figure 1. This filtering reduced the training set from 1,200 to 1,052 samples.

After filtering, we stratified the remaining data into training and internal validation splits, ensuring similar distributions of summary lengths in both. We used an 80/20 split, resulting in 841 training samples and 211 internal validation samples. Note that the official validation and test splits were kept held-out and only used for evaluation, as their reference summaries were not released.

3.2 Finetuning Pipeline

Our finetuning pipeline consists of two main stages, two-step supervised finetuning, followed by RLVR. The supervised stages adapt the base model to the task-specific summarization style, and the reinforcement step further aligns the model toward better word choice and target metrics via GSPO (Zheng et al., 2025). We use `Qwen/Qwen3-4B-Instruct-2507` (Yang et al., 2025) as the base model because of its strong instruction following capability and long context window (up to 262,144 tokens). We chose the non-thinking variant to reduce rollout cost during RL. All finetuning runs used LoRA (Hu et al., 2021) to stay within our compute budget. Training was constrained to a single A100 80GB GPU, costing about $1.39/hour.[3]

Two-Stage Supervised Finetuning Because inputs can be very long and compute is limited, we split supervised finetuning into two stages.

In the first stage, we finetuned with a max sequence length of 16,384 tokens. We applied Rank-stabilized LoRA (Kalajdzievski, 2023) to all modules[4] using LoRA rank 256 and alpha of 32. We used Adam (Kingma and Ba, 2017) with a constant learning rate of 2e-4 following (Schulman and Lab, 2025), batch size 32, and trained for two epochs. This model is reported as 'SFT Stage 1'.

In the second stage, we merged the stage-1 adapter into the base model and continued training on longer-context samples. We filtered data to those with total prompt+completion length between 10,000 and 30,000 tokens, solely improve performance towards long input summary. To fit on a single 80GB GPU at this length, we reduced the LoRA rank to 32 and attached adapters only to `up_proj` and `down_proj`. We trained for one epoch with the same learning rate and batch size as stage 1. This model is reported as 'SFT Stage 2'.

Reinforcement Tuning Finally, we applied reinforcement tuning to directly optimize summarization metrics: BLEU, ROUGE-L, and ROUGE-2. BLEU was weighted 3× higher because the SFT model already achieved strong ROUGE but comparatively lower BLEU. We used DAPO (Yu et al., 2025) with high clip value and improved rollout sampling to increase reward diversity, and we computed importance sampling at sequence level following (Zheng et al., 2025). For optimizer, we used Adam 8-bit (Dettmers et al., 2022) with constant learning rate 8e-5, max grad norm 0.2, 16 rollouts, LoRA rank 2 and alpha 1, rollout temperature 1.0, max sequence length 12,000, and clip value 0.28 as in (Yu et al., 2025). We use low LoRA rank for following (Schulman and Lab, 2025) which stated that RLVR have a sparse reward thus lower training signal, making lower LoRA rank more suitable. We used the Unsloth (Daniel Han and team, 2023) implementation for efficient RL. During RL, we observed training instabilities (see Section 4), so selected the last stable checkpoint (step 500) for evaluation. This model is reported as 'Reinforcement Tuned' in the result table.

[3] Price according to https://www.runpod.io/pricing.

[4] `q_proj`, `k_proj`, `v_proj`, `o_proj`, `gate_proj`, `up_proj`, and `down_proj`.

Model	Avg	Rouge-2	Rouge-L	BLEU
Internal Validation results				
Qwen3-4B-Thinking-2507	11.60	12.87	16.79	5.14
Qwen3-4B-Instruct-2507	16.15	18.79	22.31	7.33
SFT Stage 1	22.96	29.32	29.19	10.38
SFT Stage 2	24.55	30.89	30.49	12.29
Reinforcement Tuned	**32.71**	**35.60**	**34.43**	**28.11**
Validation Leaderboard results				
SFT Stage 1	25.47	31.25	31.42	13.74
SFT Stage 2	**25.57**	**31.51**	**31.77**	**13.43**
Test Leaderboard results				
SFT Stage 2	23.94	30.35	30.19	11.27
Reinforcement Tuned	**29.91**	**34.91**	**33.34**	**21.49**

Table 1: Performance of models across internal validation, validation leaderboard, and test leaderboard results.

4 Results & Discussion

The main results are summarized in Table 1.

Qwen3-4B Instruct outperforms the Thinking variant. From Table 1, the instruction-tuned model scores clearly higher than the thinking version on summarization. We attribute this to the nature of the task: judgments are long and information-dense, so additional chain-of-thought style reasoning may introduce token overhead and distract from concise summary generation. This observation supports our choice of `Qwen3-4B-Instruct-2507` as the base model.

Reinforcement tuning can collapse late in training. During GSPO-based reinforcement tuning, we observed occasional reward collapse, characterized by repetitive rollouts reaching the max se-

Figure 2: An observed behavior of reward collapsed when the model underwent GSPO training.

Model	Avg	Rouge-2	Rouge-L	BLEU
Internal Validation results				
SFT Stage 2	24.55	30.89	30.49	12.29
SFT Unified	22.67	29.45	29.07	9.49
Test Leaderboard results				
SFT Stage 2	27.21	33.36	32.25	16.01
SFT Unified	21.62	28.46	28.42	7.97

Table 2: Performance of the model that underwent two-stage supervised finetuning pipeline compared to unified pipeline.

quence length and a subsequent sharp drop in rewards as shown in Figure 2. We hypothesized that this collapse could resulted from an instability inference-training mismatch, which can be either from precision sensitivity (Qi et al., 2025), or hardware instability (Liu et al., 2025a). To avoid evaluating such degraded checkpoints, we selected the last stable checkpoint (step 500) for reporting. We also provide some analysis of model performance on different checkpoints on Appendix B.

Cost analysis of the best model. Our full pipeline used about 35 GPU-hours in total[5] On RunPod pricing ($1.39/hour), this corresponds to roughly $50 for end-to-end training, making the approach practical for single-GPU setups.

5 Ablation Study

To further validate the effectiveness of our approach, we conducted two ablation studies. First, we examined whether the two-stage supervised finetuning pipeline can be replaced with a single unified stage. Second, we evaluated the impact of prompt design by comparing a carefully curated prompt against a naive prompt (see Appendix A).

Two-stage supervised finetuning outperforms a unified pipeline. Because our finetuning strategy attaches different LoRA adapters across two stages, we tested whether training only with the second-stage setup (long-context, LoRA only on MLP modules) could match its performance. To do this, we finetuned `Qwen3-4B-Instruct-2507` using the stage-2 hyperparameters, increased the training epochs to 3, and removed the minimum prompt-length filter. We denote this as 'SFT Unified' in Table 2. As shown in the table, **SFT Stage 2 consistently outperforms SFT Unified**, highlighting the benefit of first adapting the model on shorter contexts with more trainable parameters before extending to long-context finetuning.

Naive prompts converge faster, but curated prompts win with longer training. We also compared two-stage finetuning under two prompts: a naive prompt with no heuristic guidance (Appendix A) and our curated summarization prompt. Both models used the same stage-1 finetuning setup described in Section 3.2. The results are summarized in Figure 3.

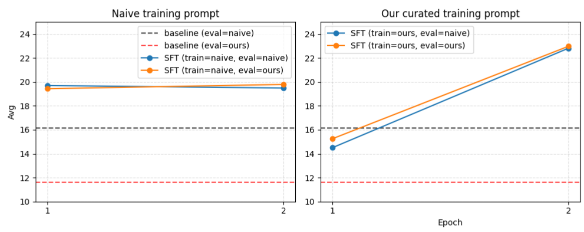


Figure 3: Plot of internal validation performance of Qwen3-4B-Instruct-2507 underwent SFT under different prompt, and evaluate on different prompt. Horizontal axis denotes training epochs and vertical axis denotes average metric among ROUGE-2, ROUGE-L, and BLEU.

We observed that the naive prompt learns faster in the early steps, reaching an average score of ∼19.5 after the first epoch, while the curated prompt lags behind at ∼14.5, slightly below baseline. However, after the second epoch, the naive-prompt model saturates and gains little, whereas the model trained with the curated prompt continues to improve beyond an average score of 22. This suggests that **well-designed summarization prompts yield better final performance when training runs longer**.

We also tested cross-prompt robustness by evaluating models on prompts different from those seen in training. Under our setup and hyperparameters, **the finetuned models did not exhibit strong sensitivity to unseen prompts**, indicating reasonable prompt generalization.

[5] 51 minutes for SFT stage 1, 31 minutes for SFT stage 2, and 33 hours for reinforcement tuning.

6 Conclusion

We presented a practical pipeline for finetuning `Qwen3-4B-Instruct-2507` on long-form legal summarization. Our approach begins with length-based data filtering to remove samples whose judgment–summary ratio deviates from the expected trend, reducing noise in the training set. We then apply a two-stage supervised finetuning strategy with LoRA, first adapting the model on shorter contexts and then extending to long-context data, followed by GSPO-based reinforcement tuning to directly optimize ROUGE and BLEU. Experiments showed that the two-stage SFT setup outperforms both the base model and a unified SFT pipeline, while RL yields the largest performance gains on internal and leaderboard evaluations. Our ablations further suggest that prompt design matters more at longer training horizons and that the model remains reasonably robust to unseen prompts. Notably, the entire recipe fits within a single 80GB GPU and costs roughly $50, demonstrating that competitive legal summarization is achievable under modest compute constraints.

Limitations

This work has several limitations. First, our evaluation of the two-stage finetuning pipeline is incomplete: we fixed the second-stage training to a single epoch and did not systematically explore longer training, curriculum-style schedules, or error breakdowns by sequence length. As a result, it is still unclear whether additional long-context finetuning would yield further gains or help specific length regimes.

Second, the reinforcement tuning phase exhibited training collapse, and we only mitigated it operationally (by selecting the last stable checkpoint) rather than fully diagnosing or fixing the underlying cause. A deeper study of rollout diversity, reward shaping, and clipping strategies would likely improve stability.

Third, we did not exhaust the RLVR design space: we did not compare against vanilla GRPO (Shao et al., 2024) without sequence-level grouping, nor did we test alternative reward configurations such as single-metric rewards versus our compound setup. These choices may affect both final quality and robustness.

Finally, the reported $50 training cost reflects only the successful run on a single 80GB GPU. When including exploratory and failed runs, the total compute was closer to 70 GPU-hours (about $100), so the true end-to-end cost of reproducing this work may be higher than the headline number.

References

Pawitsapak Akarajaradwong, Chompakorn Chaksangchaichot, Pirat Pothavorn, Attapol Thamrongrattanarit-Rutherford, Ekapol Chuangsuwanich, and Sarana Nutanong. 2025. Can group relative policy optimization improve thai legal reasoning and question answering? *Preprint*, arXiv:2507.09638.

Yapei Chang, Yekyung Kim, Michael Krumdick, Amir Zadeh, Chuan Li, Chris Tanner, and Mohit Iyyer. 2025. Bleuberi: Bleu is a surprisingly effective reward for instruction following. *Preprint*, arXiv:2505.11080.

Yukang Chen, Shengju Qian, Haotian Tang, Xin Lai, Zhijian Liu, Song Han, and Jiaya Jia. 2024. Longlora: Efficient fine-tuning of long-context large language models. *Preprint*, arXiv:2309.12307.

Michael Han Daniel Han and Unsloth team. 2023. Unsloth.

Debtanu Datta, Shubham Soni, Rajdeep Mukherjee, and Saptarshi Ghosh. 2023. MILDSum: A novel benchmark dataset for multilingual summarization of Indian legal case judgments. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5291–5302, Singapore. Association for Computational Linguistics.

Tim Dettmers, Mike Lewis, Sam Shleifer, and Luke Zettlemoyer. 2022. 8-bit optimizers via block-wise quantization. *Preprint*, arXiv:2110.02861.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Abhinav Joshi, Shounak Paul, Akshat Sharma, Pawan Goyal, Saptarshi Ghosh, and Ashutosh Modi. 2024. IL-TUR: Benchmark for Indian legal text understanding and reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11460–11499, Bangkok, Thailand. Association for Computational Linguistics.

Damjan Kalajdzievski. 2023. A rank stabilization scaling factor for fine-tuning with lora. *Preprint*, arXiv:2312.03732.

Diederik P. Kingma and Jimmy Ba. 2017. Adam: A method for stochastic optimization. *Preprint*, arXiv:1412.6980.

Frederic Kirstein, Terry Ruas, Robert Kratel, and Bela Gipp. 2024. Tell me what I need to know: Exploring

LLM-based (personalized) abstractive multi-source meeting summarization. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 920–939, Miami, Florida, US. Association for Computational Linguistics.

Haitao Li, Qian Dong, Junjie Chen, Huixue Su, Yujia Zhou, Qingyao Ai, Ziyi Ye, and Yiqun Liu. 2024. Llms-as-judges: A comprehensive survey on llm-based evaluation methods. *Preprint*, arXiv:2412.05579.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Dongqi Liu and Vera Demberg. 2024. RST-LoRA: A discourse-aware low-rank adaptation for long document abstractive summarization. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2200–2220, Mexico City, Mexico. Association for Computational Linguistics.

Jiacai Liu, Yingru Li, Yuqian Fu, Jiawei Wang, Qian Liu, and Yu Shen. 2025a. When speed kills stability: Demystifying rl collapse from the inference-training mismatch.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. 2025b. Understanding r1-zero-like training: A critical perspective. *Preprint*, arXiv:2503.20783.

George Michalopoulos, Kyle Williams, Gagandeep Singh, and Thomas Lin. 2022. MedicalSum: A guided clinical abstractive summarization model for generating medical reports from patient-doctor conversations. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4741–4749, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *Preprint*, arXiv:2203.02155.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Penghui Qi, Zichen Liu, Xiangxin Zhou, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. 2025. Defeating the training-inference mismatch via fp16. *Preprint*, arXiv:2510.26788.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Preprint*, arXiv:2305.18290.

John Schulman and Thinking Machines Lab. 2025. Lora without regret. *Thinking Machines Lab: Connectionism*. Https://thinkingmachines.ai/blog/lora/.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *Preprint*, arXiv:1707.06347.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *Preprint*, arXiv:2402.03300.

Saloni Sharma, Surabhi Srivastava, Pradeepika Verma, Anshul Verma, and Sachchida Nand Chaurasia. 2023. A comprehensive analysis of indian legal documents summarization techniques. *SN Comput. Sci.*, 4(5).

Abhay Shukla, Paheli Bhattacharya, Soham Poddar, Rajdeep Mukherjee, Kripabandhu Ghosh, Pawan Goyal, and Saptarshi Ghosh. 2022. Legal case document summarization: Extractive and abstractive methods and their evaluation. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1048–1064, Online only. Association for Computational Linguistics.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Jianxin Yang. 2023. Longqlora: Efficient and effective method to extend context length of large language models. *Preprint*, arXiv:2311.04879.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, and 16 others. 2025. Dapo: An open-source llm reinforcement learning system at scale. *Preprint*, arXiv:2503.14476.

Haopeng Zhang, Philip S. Yu, and Jiawei Zhang. 2024. A systematic survey of text summarization: From statistical methods to large language models. *Preprint*, arXiv:2406.11289.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. 2025. Group sequence policy optimization. *Preprint*, arXiv:2507.18071.

A Prompts

The system prompt was outlined as follows:

System Prompt

You are a precise summarizer for legal materials. Follow the user "Instruction" steps exactly.

For instruction prompt, we sampled different summary item from the training sample and extract key patterns found in the summarization. The prompt is outlined below.

Instruction Prompt

Instruction
You are given an Indian court judgment in English to summarize. Your task is to summarize the judgment based on the following strategies on summarizing the judgment:
1) Identify the authority (court, commission, regulator, ministry or parliamentary body) and its action in the opening clause – name the institution and state what it did (granted/-denied/quashed/stayed/notified/imposed, etc.), adding the date if given.
2) Summarise the parties and the core issue in one sentence – who is involved (petitioner, accused, complainant, regulator) and what the underlying dispute/offence/claim is, keeping background facts concise.
3) Mention the legal basis or tests only when they appear in the source – specify Acts, Sections, Articles, Rules or Regulations exactly as provided; never invent citations if none are given.
4) Convey the key reasoning and observations succinctly – what the bench or adjudicating authority held/observed/noted and why, paraphrasing where possible and using short quoted phrases only when present.
5) Finish with the outcome and operative directions – note the practical effect (e.g. bail granted/refused, penalty imposed, matter stayed), any deadlines, next-hearing dates or compliance steps. If the source lists multiple directions, collapse them into one sentence separated by semicolons rather than using bullets or separate lines.

Style & constraints:
- Neutral, factual tone in past tense and active voice.
- Prefers in paragraphic format not in bullet points.
- Do not include headings, bullet points or commentary.
- Preserve names, numbers and dates exactly; if a detail is absent in the source, simply omit it rather than guessing.

Input Judgment
`{{judgment}}`

Output Summary:

We also provide our naive summary prompt here.

Naive Summary Prompt

Summarize the following judgement:

Input Judgment:
`{{judgment}}`

Output Summary:

B Performance of Models Under Reinforcement Tuning Across Training Steps

In Section 4, we discussed that the reinforcement tuning stage sometimes exhibited a *collapse* of the reward signal as training progressed. To better understand this phenomenon, we conducted two runs that differed only in LoRA rank and learning rate.

The first run (denoted as `GSPO_r1_1e-4` in Table 3) followed the hyperparameters in Section 3.2, except that we used a LoRA rank of 1 and a learning rate of 1e-4. The second run (our best-performing configuration), denoted as `GSPO_r2_8e-5`, followed the settings in Section 3.2 with a higher LoRA rank 2 and a lower learning rate 8e-5.

In Figure 2, which plots the reward over training steps, `GSPO_r1_1e-4` corresponds to the red line, while `GSPO_r2_8e-5` corresponds to the orange line. We observe that `GSPO_r1_1e-4` collapses relatively early, around checkpoint 190. To improve stability, we therefore reduced the learning rate and increased the LoRA rank. This second run was indeed more stable, but it still eventually collapsed at around the 520th step.

To check whether the model's *task performance* actually improved before the collapse, we saved a checkpoint every 50 steps and evaluated each checkpoint on our internal validation split. The results are shown in Table 3.

Model	Ckpt	Avg	ROUGE-2	ROUGE-L	BLEU
Internal validation results					
GSPO_r1_1e-4	50	26.29	32.15	31.64	15.09
GSPO_r1_1e-4	100	28.60	33.79	32.53	19.47
GSPO_r1_1e-4	150	28.36	33.95	32.89	18.26
GSPO_r2_8e-5	300	31.50	35.59	33.99	24.92
GSPO_r2_8e-5	400	31.11	35.75	33.97	23.61
GSPO_r2_8e-5	500	32.71	35.60	34.43	28.11
Test leaderboard results					
GSPO_r1_1e-4	150	27.21	33.36	32.25	16.01
GSPO_r2_8e-5	500	29.91	34.91	33.34	21.49

Table 3: Internal validation and test leaderboard performance across different reinforcement tuning steps. "Avg" is the average of the reported metrics.

Overall, we observe an upward trend in downstream metrics as training progresses, despite the reward curve eventually collapsing. For example, in the more stable run (`GSPO_r2_8e-5`), BLEU increases from 24.92 at step 300 to 28.11 at step 500 on the internal validation split, and the corresponding test leaderboard score at step 500 is 21.49. At the same time, ROUGE-2 and ROUGE-L either improve slightly or remain in a similar range across checkpoints. These results suggest that the apparent instability in the reward signal is not primarily caused by degraded task learning, but is more consistent with other factors (e.g., inference–training mismatch) discussed in Section 4.

SCaLAR_NITK @ JUSTNLP Legal Summarization (L-SUMM) Shared Task - Rhetorical Role based Abstractive Hierarchical Summarization of Indian Legal Documents

Arjun T D and Anand Kumar Madasamy
Department of Information Technology
National Institute of Technology Karnataka (NITK), Surathkal, India
{arjuntd.243it001, m_anandkumar}@nitk.edu.in

Abstract

This paper presents the systems we submitted to the JUST-NLP 2025 Shared Task on Legal Summarization (L-SUMM). Creating abstractive summaries of lengthy Indian court rulings is challenging due to transformer token limits. To address this problem, we compare three systems built on a fine-tuned Legal Pegasus model. System 1 (Baseline) applies a standard hierarchical framework that chunks long documents using naive token-based segmentation. System 2 (RR-Chunk) improves this approach by using a BERT-BiLSTM model to tag sentences with rhetorical roles (RR) and incorporating these tags (e.g., [Facts]...) to enable structurally informed chunking for hierarchical summarization. System 3 (WRR-Tune) tests whether explicit importance cues help the model by assigning importance scores to each RR using the geometric mean of their distributional presence in judgments and human summaries, and finetuning a separate model on text augmented with these tags (e.g., [Facts, importance score 13.58]). A comparison of the three systems demonstrates the value of progressively adding structural and quantitative importance signals to the model's input.

1 Introduction

Automatic text summarization of legal documents is a critical, high-impact challenge in applied NLP. It offers the potential to help legal professionals quickly distill lengthy and complex case judgments, thereby improving judicial efficiency (Shukla et al., 2022). In a multilingual nation like India, this task is further complicated by the need to ensure access to justice across different languages (Datta et al., 2023). As the volume of legal text continues to grow, the development of robust benchmarks and models for the Indian legal domain has become an active area of research (Joshi et al., 2024).

The JUST-NLP 2025 Legal Summarization (L-SUMM) shared task provides a key benchmark for this problem, focusing on the abstractive summarization of Indian court judgments. A primary difficulty in this task, as noted by Sharma et al. (2023), is the extreme length of legal documents, which often exceeds the input-token limitations of modern transformer models like PEGASUS. This necessitates intelligent strategies beyond naive truncation.

A promising avenue for handling long documents is to leverage their inherent logical structure. Prior work has shown the value of semantic segmentation of legal texts (Kalamkar et al., 2022). A powerful way to represent this structure is through the identification of rhetorical roles (e.g., *Facts*, *Reasoning*, *Decision*), a technique that has been successfully applied to legal texts for summarization and analysis (Saravanan et al., 2008; Bhattacharya et al., 2019; Malik et al., 2022).

In this paper, we present our three systems submitted to the L-SUMM task, which explore a progressive integration of this structural information. Our baseline system uses a standard hierarchical, token-based chunking method. Our second system introduces a more context-aware hierarchical approach, using "rhetorical chunks" based on semantic roles. Our final, most advanced system fine-tunes a model on text embedded with data-driven importance scores for each rhetorical role, explicitly teaching the model to weigh information based on our analysis of the summary-generation process.

2 Methodology

2.1 Base Model and Fine-Tuning

All three systems leverage the **Legal Pegasus** (`nsi319/legal-pegasus`) model, chosen for its fine-tuning on legal text. We fine-tuned this model on the full 1200-document **InLSum** training set. Due to GPU memory constraints with long sequences, we employed memory-saving techniques:

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 121–126
December 24, 2025 ©2025 Association for Computational Linguistics

the Adafactor optimizer, a per-device batch size of 1, and gradient accumulation steps of 8 (effective batch size of 8). The learning rate was set to 2×10^{-5}.

2.2 Handling Document Length

A primary challenge in legal summarization is the extreme length of judgments, often exceeding the 1024-token limit of models like PEGASUS. We implemented a two-pronged strategy:

- **Short Documents:** Judgments approximated as shorter than 1024 tokens were summarized directly by feeding the entire text to the model.
- **Long Documents:** Judgments exceeding the threshold were processed using a hierarchical summarization approach, detailed differently for each system below.

2.3 System 1: Baseline Hierarchical Summarization

Our baseline system addresses the challenge of long documents using a standard hierarchical summarization technique, illustrated in Figure 1. For documents exceeding the model's input limit (approximated by a 4096-character threshold), the text is first divided into overlapping chunks of approximately 900-1000 tokens. This naive token-based splitting often disrupts the natural semantic flow of the legal text. Each chunk is then summarized independently by our fine-tuned Legal Pegasus model, capturing primarily local context. These initial summaries are recursively combined in pairs (or small groups) and re-summarized, building a tree structure where each ascending level incorporates context from a wider portion of the original document. This step is repeated recursively until the final summary is generated, which will have context of all the initial chunks.

2.4 System 2: Hierarchical Summarization with Rhetorical Chunking (RR-Chunk)

System 2 enhances the hierarchical approach by incorporating semantic structure. We first preprocess the entire dataset using a BERT-BiLSTM model, which uses BERT model which is finetuned on Indian legal corpus for word embeddings and two Bi-LSTMs for sentence and document level context followed by a classification head to tag each sentence with one of seven rhetorical roles (e.g., *Facts*, *Decision*). A Legal Pegasus model is then fine-tuned on this enriched data, learning to recognize text prepended with role tags (e.g., `[Facts] The petitioner...`). During inference for long documents, instead of splitting by token count, we employ **rhetorical chunking**: consecutive sentences sharing the same role are grouped into a single chunk. This preserves the logical units of the judgment (like keeping all facts together) and provides more coherent segments to the summarizer. But if the chunk itself is longer than 1024 tokens then the chunk is again split using a 900-1000 token threshold. The same multi-level hierarchical summarization process depicted in Figure 1 is then applied, using these semantically meaningful rhetorical chunks as the base units and the RR-tuned model for summarization at each level. This approach aims to guide the language model more effectively by leveraging the inherent structure of the legal document, hypothesizing that summaries generated from complete rhetorical units will be superior. Short documents are summarized directly using the RR-tuned model.

2.5 System 3: Fine-Tuning with Weighted Rhetorical Roles (WRR- Tune)

Our third system investigates whether explicitly signaling the data-driven importance of each rhetorical role during fine-tuning can further enhance summarization.

Algorithm 1 RR Importance Scoring

Require: Tagged Judgments J, Tagged Summaries S
Initialize $C_j[r] \leftarrow 0$, $C_s[r] \leftarrow 0$ for all roles r
Initialize $T_j \leftarrow 0$, $T_s \leftarrow 0$
for all document d in $J \cap S$ **do**
    for all sentence $sent$ in d_{judg} **do**
        $r \leftarrow$ get_role($sent$)
        $C_j[r] \leftarrow C_j[r] + 1$; $T_j \leftarrow T_j + 1$
    end for
    for all sentence $sent$ in d_{summ} **do**
        $r \leftarrow$ get_role($sent$)
        $C_s[r] \leftarrow C_s[r] + 1$; $T_s \leftarrow T_s + 1$
    end for
end for
for all r in all unique roles **do**
    $P_j \leftarrow (C_j[r]/T_j) \times 100$
    $P_s \leftarrow (C_s[r]/T_s) \times 100$
    $Ret \leftarrow P_s/P_j$
    $Score[r] \leftarrow \sqrt{Ret \times P_s}$
end for
return $Score$ sorted descending

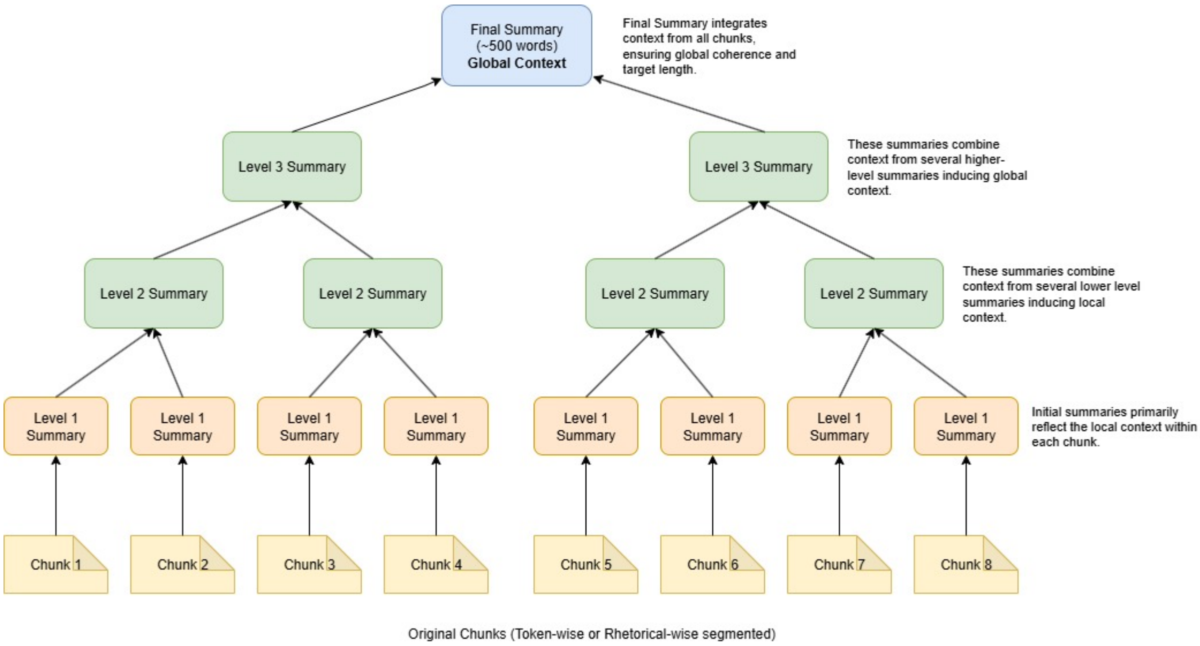


Figure 1: Hierarchical summarization tree. Initial summaries (Level 1) capture local context from base chunks (token-based or rhetorical). Subsequent levels combine these summaries, progressively incorporating broader context. until the final summary integrates information globally.

Algorithm 1 estimates how important each rhetorical role is to human-written summaries. Let J denote the set of tagged judgments and S the set of tagged summaries. For every document d that appears in both sets ($d \in J \cap S$), we iterate over all judgment sentences d_{judg} and summary sentences d_{summ}, and obtain their rhetorical roles. We maintain two role-count distributions: $C_j[r]$ records how many judgment sentences belong to role r, and $C_s[r]$ records the same for summaries. We also track total sentence counts T_j and T_s across judgments and summaries respectively. After accumulating these counts over all documents, we compute the percentage frequency of each role in judgments as $P_j = (C_j[r]/T_j) \times 100$ and in summaries as $P_s = (C_s[r]/T_s) \times 100$. The retention ratio $Ret = P_s/P_j$ captures how strongly role r is preserved from judgment to summary. Finally, the overall importance score for each role is given by $Score[r] = \sqrt{Ret \times P_s}$, which emphasizes roles that are both frequently included in summaries and retained at a high rate. Roles are then ranked in descending order of $Score[r]$.

We first derived an importance score for each rhetorical role based on its representation in the training data, as detailed in Algorithm 1. The core idea is that roles significantly more concentrated in human-written summaries (relative to their presence in full judgments) are more important for summarization . To balance this **retention** factor with the role's absolute **presence (volume)** in the summary, we employed the **geometric mean** (`sqrt(Retention * Summary percentage(volume)`), ensuring high scores are assigned only to roles that are both highly retained and substantially present. This analysis yielded the scores shown in Table 1, identifying roles like *Facts* and *Decision* as most important. The seven rhetorical roles are Facts, Reasoning, None, Decision, Arg Petitioner, Arg Respondent and Issue. As show in table 1, each sentence is classified into one of the seven rhetorical roles along with it's importance score. We then created a new version of the training dataset where the input text embedded these scores within the role tag using a human-readable format, for example, `[Facts, importance score 9.48] The petitioner....` This descriptive tag provides a clearer linguistic signal to the model about the score's meaning compared to just embedding rhetorical role. A separate Legal Pegasus model (**WRR-Tune**) was then fine-tuned from scratch on this new weighted-role dataset, learning to directly associate the explicit importance score with the summarization task during training. For inference, System 3 uses the same **rhetorical chunking** hierarchical method as System 2, but utilizes this specialized WRR-Tuned model to generate summaries at each level, thus leveraging the

Role	Imp Score	Retention	Summary %
Facts	9.48	2.83x	31.73%
Reasoning	5.61	2.18x	14.43%
None	4.93	0.64x	37.87%
Decision	4.77	3.60x	6.33%
Arg. Petitioner	2.09	0.63x	6.92%
Arg. Respondent	0.73	0.23x	2.28%
Issue	0.60	0.81x	0.44%

Table 1: Rhetorical Role Importance Scores.

learned importance weights throughout the process. This system tests whether the model can effectively learn and utilize explicit, data-driven importance signals provided directly within the input during fine-tuning.

3 Results and Discussions

3.1 Computational Environment

All experiments were conducted using a virtual machine equipped with an Intel(R) Xeon(R) CPU @ 2.00GHz and an NVIDIA Tesla P100 GPU with 16GB VRAM. The models were implemented in Python using the PyTorch deep learning framework, along with the Hugging Face Transformers library for BERT-based architectures. All the experiments being reported in the paper including the comparative studies were done by us, in this computational setup.

3.2 Results

The performance of our three systems on the validation set and the performance of System 3 on the test set are presented in Table 2.

As shown in Table 2, there is a clear and consistent improvement across all metrics on the validation set as we progressed from System 1 to System 3. System 2 (RR-Chunk), which utilized rhetorical roles for more coherent chunking, significantly outperformed the baseline System 1, highlighting the benefit of incorporating semantic structure into the hierarchical process. System 3 (WRR-Tune), which fine-tuned the model with explicit, data-driven importance scores embedded in the input, achieved the best performance on the validation set by a considerable margin, particularly demonstrating strong gains in ROUGE-2 and BLEU scores. This confirms our hypothesis that providing the model with quantitative importance signals during fine-tuning is a highly effective strategy for legal summarization. On the final test set, System 3 maintained strong performance, achieving an average score of 20.74 and a ROUGE-L score of 25.93.

Overall, the steady rising trend in all three systems indicates that summarization performance is greatly improved by combining both structure information (by rhetorical roles) and quantitative salience cues (via significance ratings). The advancements show that models that are informed by both explicit markers of content relevance and linguistic structure are beneficial for legal abstraction.

3.3 Discussion

The results clearly demonstrate a progressive improvement from System 1 to System 3 across all evaluation metrics. System 1, which relies on a standard hierarchical summarization pipeline with naive token-based chunking, provides a reasonable baseline but is limited by its inability to preserve the semantic structure of legal documents. As a result, the model often receives context fragments that do not align with coherent discourse units, reducing the effectiveness of the hierarchical encoder–decoder process.

System 2 (RR-Chunk) shows a noticeable increase in ROUGE-2, ROUGE-L, and BLEU, which highlights the advantage of using rhetorical roles for segmentation. Since Indian court judgments follow a well-defined argumentative structure, grouping text according to rhetorical roles leads to more meaningful chunks. This enables the model to better capture fact-heavy and decision-relevant sections, improving the overall quality of the generated summaries.

System 3 (WRR-Tune) achieves the highest performance on both the validation and test sets. By fine-tuning the model with explicit importance scores embedded directly into the input, the system gains an additional signal that helps it prioritize legally salient content during generation. These importance cues guide the model toward focusing on segments that contribute more substantially to accurate and coherent summaries. The stronger gains in ROUGE-2 and BLEU suggest that importance-weighted fine-tuning enhances the model's ability to reproduce key multiword expressions and legally significant phrasing.

Overall, the consistent upward trend across all three systems confirms that incorporating both structural information and quantitative salience cues significantly boosts summarization performance. The improvements indicate that legal abstraction benefits from models that are guided not only by linguistic structure but also by explicit indicators of content relevance.

Dataset	System	AVG	ROUGE-2	ROUGE-L	BLEU
Validation Set Results					
Validation	System 1 (Baseline Hierarchical)	18.65	18.81	24.43	12.70
Validation	System 2 (RR-Chunk)	19.93	20.37	25.16	14.26
Validation	**System 3 (WRR-Tune)**	**21.53**	**22.57**	**26.28**	**15.75**
Test Set Results					
Test	**System 3 (WRR-Tune)**	20.74	21.86	25.93	14.43

Table 2: Performance comparison of the three systems on the InLSum validation set and the final test set performance for System 3. Scores are reported as provided by the shared task organizers.

4 Conclusion and Future Work

In this paper, we presented three systems for the JUST-NLP 2025 Legal Summarization shared task, all based on a fine-tuned Legal Pegasus model. Our methods progressed from a standard hierarchical baseline (System 1) to a semantically-aware model using rhetorical-role-based chunking (System 2), and finally to a novel system fine-tuned on text embedded with data-driven importance scores (System 3). Our experiments on the validation set show a clear and consistent performance improvement at each stage, with System 3 achieving the highest scores across all metrics. This confirms our hypothesis that progressively enriching the model's input with both structural-semantic information (rhetorical roles) and quantitative, data-driven signals (importance scores) is a highly effective strategy for producing more accurate and coherent summaries of complex legal judgments. For future work, we plan to address the limitations of hierarchical chunking by experimenting with end-to-end long-context models such as LED or Long-T5, which can process entire documents at once. We also plan to work on explainablity and ensembling with other well performing LLMs like BART and LED.

Limitations

The hierarchical summarization framework, used in all three systems, is a necessary workaround but is not ideal. It still risks context loss at chunk boundaries and, more significantly risks the coherency of the final summary . Furthermore, the performance of Systems 2 and 3 is fundamentally dependent on the accuracy of the upstream BERT-BiLSTM model used for rhetorical role tagging. Any errors from this classifier are propagated and potentially amplified by the summarization model, which has been trained to trust these (sometimes incorrect) structural and weighted tags.

Ethics Statement

This research was conducted using publicly available legal dataset released for academic and research purposes. No private or personally identifiable information was involved at any stage. The primary goal of this work is to explore abstractive summarization for legal documents. While the proposed models show promising results, they reflect patterns present in the training data. Any biases, inaccuracies, or limitations in the dataset may influence model predictions. Therefore, these models should not be seen as replacements for human legal reasoning. We strongly encourage users to apply this work responsibly and ethically, keeping in mind the sensitive nature of legal decision making.

Acknowledgments

This work was supported by the ANRF-SERB-CRG project titled "A Deep Explainable Framework for Semantically Similar Document Retrieval and Summarization of Legal Text," under the supervision of Dr. Anand Kumar M, Department of Information Technology, National Institute of Technology Karnataka, Surathkal.

References

Paheli Bhattacharya, Shounak Paul, Kripabandhu Ghosh, Saptarshi Ghosh, and Adam Wyner. 2019. Identifying rhetorical roles of sentences in indian legal judgments. In *Proceedings of the 32nd International Conference on Legal Knowledge and Information Systems (JURIX)*.

Debtanu Datta, Shubham Soni, Rajdeep Mukherjee, and Saptarshi Ghosh. 2023. MILDSum: A novel benchmark dataset for multilingual summarization of Indian legal case judgments. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5291–5302, Singapore. Association for Computational Linguistics.

Abhinav Joshi, Shounak Paul, Akshat Sharma, Pawan Goyal, Saptarshi Ghosh, and Ashutosh Modi. 2024. IL-TUR: Benchmark for Indian legal text understanding and reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11460–11499, Bangkok, Thailand. Association for Computational Linguistics.

Prathamesh Kalamkar, Aman Tiwari, Astha Agarwal, Saurabh Karn, Smita Gupta, Vivek Raghavan, and Ashutosh Modi. 2022. Corpus for automatic structuring of legal documents. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*.

Vijit Malik, Rishabh Sanjay, Shouvik Kumar Guha, Angshuman Hazarika, Shubham Nigam, Arnab Bhattacharya, and Ashutosh Modi. 2022. Semantic segmentation of legal documents via rhetorical roles. In *Proceedings of the Natural Legal Language Processing Workshop 2022*, pages 153–171. Association for Computational Linguistics.

M. Saravanan, B. Ravindran, and S. Raman. 2008. Automatic identification of rhetorical roles using conditional random fields for legal document summarization. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-I*. IJCNLP.

Saloni Sharma, Surabhi Srivastava, Pradeepika Verma, Anshul Verma, and Sachchida Nand Chaurasia. 2023. A comprehensive analysis of Indian legal documents summarization techniques. *SN Computer Science*, 4(5).

Abhay Shukla, Paheli Bhattacharya, Soham Poddar, Rajdeep Mukherjee, Kripabandhu Ghosh, Pawan Goyal, and Saptarshi Ghosh. 2022. Legal case document summarization: Extractive and abstractive methods and their evaluation. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1048–1064, Online only. Association for Computational Linguistics.

goodmen @ L-MT Shared Task: A Comparative Study of Neural Models for English-Hindi Legal Machine Translation

Deeraj S K, Karthik Suryanarayanan, Yash Ingle, Pruthwik Mishra
Sardar Vallabhbhai National Institute of Technology, Surat
{u23ai050, u23cs013, u23ai062}@coed.svnit.ac.in
{pruthwikmishra}@aid.svnit.ac.in

Abstract

In a massively multilingual country like India, providing legal judgments in understandable native languages is essential for equitable justice to all. The Legal Machine Translation (L-MT) shared task focuses on translating legal content from English to Hindi which is the most spoken language in India. We present a comprehensive evaluation of neural machine translation models for English-Hindi legal document translation, developed as part of the L-MT shared task. We investigate four multilingual and Indic focused translation systems. Our approach emphasizes domain specific fine-tuning on legal corpus while preserving statutory structure, legal citations, and jurisdictional terminology. We fine-tune two legal focused translation models, InLegalTrans and IndicTrans2 on the English-Hindi legal parallel corpus provided by the organizers where the use of any external data is constrained. The fine-tuned InLegalTrans model achieves the highest BLEU score of 0.48. Comparative analysis reveals that domain adaptation through fine-tuning on legal corpora significantly enhances translation quality for specialized legal texts. Human evaluation confirms superior coherence and judicial tone preservation in InLegalTrans outputs. Our best performing model is ranked 3rd on the test data.

1 Introduction

Legal translation is one of the most challenging domains in natural language processing, requiring not only linguistic accuracy, but also preservation of legal semantics, statutory structure, and jurisdictional terminology. In multilingual legal systems such as India's, where legal proceedings and documentation occur across multiple languages, accurate translation between English and Indian languages is essential for ensuring access to justice and legal transparency. The linguistic and cultural gap between English legal texts and their Hindi translations demands specialized translation systems that can handle domain specific terminology, complex sentence structures, and formal register. Recent studies in legal NLP highlight concrete failures of general-purpose systems on legal discourse: domain-specific pretraining or fine-tuning consistently improves performance on legal tasks such as judgment classification, statutory retrieval, and terminology preservation (Chalkidis et al., 2020; Zheng et al., 2021; Chu and Wang, 2018). For Indic language pairs, large parallel resources such as Samanantar have enabled improved base models for Indian languages (Ramesh et al., 2021), yet English–Indic legal translation remains underresourced compared to English–European pairs. This gap is further amplified by code-switching and transliteration phenomena in Indian legal texts, which complicate tokenization and lexical alignment (see e.g. (Mujadia et al., 2024)).

Empirical evidence from prior MT research indicates that domain adaptation, either via continued pre-training on in-domain corpora or targeted fine-tuning—yields substantial gains in adequacy and terminology fidelity compared to out-of-domain baselines (Chu and Wang, 2018; Farajian et al., 2017; Rossi and Chevrot, 2019). Furthermore, recent large multilingual models (e.g., NLLB-200) demonstrate strong cross-lingual transfer but often underperform specialized, domain-adapted models on niche corpora unless further adapted (Costa-jussà et al., 2022; Mahapatra et al., 2025).

Traditional rule-based and statistical machine translation approaches have historically struggled with the specialized vocabulary and syntactic complexity present in legal documents. The advent of neural machine translation (NMT) (Sutskever et al., 2014; Bahdanau et al., 2015; Vaswani et al., 2017) has significantly improved translation quality across general domains, yet legal translation remains underexplored, particularly for low-resource language pairs such as English-Hindi in

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 127–132
December 24, 2025

legal contexts.

The L-MT shared task on English-Hindi legal translation provides a standardized evaluation framework for developing translation systems in the legal domain. This addresses the critical need for automated translation tools capable of processing Indian legal judgments, statutes, and legal documents while maintaining semantic fidelity and legal phrasing accuracy. In this work, we present a comprehensive evaluation of four translation systems spanning different architectural paradigms and parameter scales: IndicTrans2 (200M), a specialized encoder-decoder model for Indic languages; InLegalTrans (1B), a domain-adapted model pre-trained on legal corpora; NLLB-200 Distilled (1.3B), a massively multilingual baseline; and Gemini 2.0 Flash API, a large-scale commercial model. Our investigation focuses on the impact of domain-specific fine-tuning, legal-aware preprocessing, and terminology preservation strategies on translation quality.

2 Related Work

Neural machine translation has evolved significantly since the introduction of attention-based sequence-to-sequence architectures (Bahdanau et al., 2015) and self-attention based transformer models (Vaswani et al., 2017). Pre-trained multilingual models such as mBART (Liu et al., 2020), mT5 (Xue et al., 2021), and NLLB-200 (Costa-jussà et al., 2022) have demonstrated impressive zero-shot translation capabilities across hundreds of languages through large-scale multilingual pre-training. These models leverage cross-lingual transfer learning to achieve robust performance even on low-resource language pairs.

For Indic languages, the Samanantar corpus (Ramesh et al., 2021) provided large-scale parallel data for English to Indic and Indic to Indic language pairs that led to the development of IndicTrans. This model introduced the first large-scale model specifically designed for Indian language pairs. IndicTrans2 (Gala et al., 2023) extended this work with improved architectures, larger training corpora, better noise filtering, and better handling of script normalization. Recent parameter-efficient fine-tuning methods including LoRA (Hu et al., 2021) and adapter layers (Houlsby et al., 2019) have enabled domain adaptation with minimal computational overhead retaining pre-trained knowledge.

Although all these NMT systems have been seamlessly integrated into many domains through domain adaptation, legal translation poses unique challenges including specialized terminology, formal register, and complex syntactic structures characteristic of statutory language (Cao, 2007; Šarčević, 2000). Domain-specific models such as Legal-BERT (Chalkidis et al., 2020) and legal domain pre-training approaches (Zheng et al., 2021) have demonstrated the value of legal domain pre-training for natural language understanding of English legal text. However, legal NMT for English-Indic language pairs remains critically underexplored despite the practical importance in multilingual legal systems.

Prior work on legal NMT has focused on domain adaptation through continued pre-training on legal corpora (Chu and Wang, 2018; Farajian et al., 2017) and incorporation of legal terminology glossaries (Rossi and Chevrot, 2019). Our work extends this research by conducting a comparative evaluation of multiple neural architectures for English-Hindi legal translation, demonstrating substantial quality improvements through domain-specific fine-tuning while maintaining strict shared-task constraints that prohibit external data usage. MILPac (Mahapatra et al., 2025) consists of MT benchmarks in the legal domain vetted by law practioners for several English and low resource Indian language pairs. The authors also released InLegalTrans, an multilingual NMT model fine-tuned on IndicTrans2. LMs (Zhu et al., 2024) also have been widely used for machine translation where the machine translation is performed through a decoder only model rather than the traditional encoder-decoder models. The translation capabilities from English to diverse Indian languages (Mujadia et al., 2024) of different LLMs have been studied.

3 Methodology

3.1 Dataset

The L-MT English-Hindi Legal Translation corpus (Singh et al., 2025) consists of parallel sentence pairs extracted from Indian legal documents, including court judgments, statutory provisions, and legal proceedings. Each entry contains an English source sentence and its corresponding Hindi translation in Devanagari script. The corpus exhibits domain-specific characteristics typical of Indian legal discourse, including complex syntactic structures, specialized terminology, and formal register. It includes 50,000 samples provided for fine-tuning, 5,000 samples provided for validation during the training phase (validation set), and 5,000 samples on which the final BLEU score was calculated (test set).

The dataset preserves legal-specific elements including citation patterns (e.g., "AIR 1997 SC 1234", "Section 125 of the CrPC"), constitutional articles, domain-specific legal terminology such as "writ petition", "respondent", "jurisdiction" and "fundamental rights" that require accurate translation or appropriate transliteration, and numerical consistency with dates, case numbers, section identifiers, and monetary amounts preserved across source and target sentences. For submissions and evaluations, we train on the complete training corpus to maximize model exposure to domain-specific patterns.

Table 1: Corpus statistics for English-Hindi legal translation.

Split	# Samples	Usage
Training	50,000	Fine-tuning
Validation	5,000	Dev evaluation
Test	5,000	Final evaluation

The training corpus exhibits linguistic diversity across legal sub-domains including constitutional law, criminal procedure, civil litigation, contract law, and property law. Average sentence length is approximately 28 tokens for English and 32 tokens for Hindi, reflecting the morphological richness of Devanagari script. The corpus maintains authentic translation challenges including ambiguous legal terminology, code-switching patterns where English legal terms are retained in Hindi translations, and syntactic divergence in terms of grammatical structures. All experiments strictly adhere to shared-task constraints by using only the official provided datasets without external corpora, back-translation, or synthetic augmentation.

3.2 Training Details

The models are trained for 3 epochs with batch size 4 and learning rate 2e-4 on an NVIDIA A100 GPU with 94GB RAM. We follow IndicTrans2's preprocessing guidelines for normalization and tokenization. Script tags in the format Eng_Latn $\rightarrow$ Hin_Deva are added to each translation pair to ensure script consistency.

Table 2: Training configuration for InLegalTrans-en2Indic-1B

Component	Setting
Base model	InLegalTrans-en2Indic-1B
Quantization	4-bit NF4 (bitsandbytes)
LoRA rank/alpha	$r = 16$, $\alpha = 32$
Target modules	q_proj, k_proj, v_proj, o_proj, fc1, fc2
Optimizer	paged_adamw_32bit
Learning rate	2×10^{-4} (linear decay)
Batch size	4 per device
Epochs	3
Precision	FP16 mixed precision
Max seq. length	512 tokens
Hardware	NVIDIA A100 (94 GB)
Notes	Gradient accumulation used to simulate larger batch size

Particular care is taken to preserve legal symbols, citations, and numbering. Symbols like "§," "Sec.," and "Art." are kept exactly as they are. We eliminate pairs with empty target translations and filter out sentence pairs with significant length mismatches (greater than 3:1) to avoid model confusion. To minimize GPU memory consumption and enable efficient domain adaptation, InLegalTrans is optimized with parameter-efficient adapters using LoRA/PEFT techniques.

3.3 Evaluation Metrics

We evaluate the translation quality using three complementary metrics: BLEU (Papineni et al., 2002) for n-gram precision, ROUGE-L (Lin, 2004) for longest common subsequence to assess sentence-level structural preservation, and chrF++ (Popović, 2017) for character-level or subword accuracy. For inference, we use beam search decoding with beam width of 4 and temperature of 0.7, followed by post-processing for punctuation restoration and formatting corrections.

4 Results and Analysis

Table 3 shows detailed evaluation metrics across all three metrics on the final test set.

4.1 Quantitative Analysis

From the results, we observe that the fine-tuned InLegalTrans model achieves the highest BLEU score of 0.48, representing a 55% improvement over the base model (0.31 BLEU). This model

Table 3: Detailed evaluation metrics on test set. (final leaderboard score)

Model	BLEU	ROUGE-L	chrF++
Base Models			
IndicTrans2 (Gala et al., 2023)	0.30	0.42	0.52
Gemini 2.0 Flash (few-shot)	0.16	0.35	0.43
InLegalTrans (Mahapatra et al., 2025)	0.31	0.44	0.54
NLLB-200 (Costa-jussà et al., 2022)	0.27	0.39	0.49
Fine-tuned Models			
IndicTrans2 (Gala et al., 2023)	0.31	0.43	0.53
InLegalTrans (Mahapatra et al., 2025)	**0.48**	**0.56**	**0.73**

demonstrates superior performance in preserving the inherent characteristics of the legal texts. IndicTrans2 maintains stable performance at approximately 0.31 BLEU regardless of fine-tuning, suggesting strong pre-trained generalization to formal text but limited domain adaptation capacity for legal-specific patterns.

Several key observations emerge from the evaluation:

Domain Adaptation Impact: Fine-tuning on legal corpora yields substantial improvements only for InLegalTrans (+0.17 BLEU), while IndicTrans2 showed minimal gains (+0.01 BLEU). This suggests that InLegalTrans's architecture and legal pre-training approach are more receptive to domain-specific adaptation.

Model Scale vs. Specialization: Despite having fewer parameters (1B vs. 1.3B), the fine-tuned InLegalTrans significantly outperformed the larger NLLB-200 model (0.48 vs. 0.27 BLEU). This demonstrates that domain specialization and targeted fine-tuning can be more effective than raw model capacity for specialized translation tasks.

Commercial Model Performance: Gemini 2.0 Flash achieved the lowest BLEU score (0.16) despite being a large-scale commercial model. Manual inspection reveals that while Gemini produces fluent translations, they frequently deviate from legal phrasing conventions and exhibited semantic inconsistencies in handling statutory language, instead preferring generic terms in generation.

Metric Consistency: Performance rankings remains consistent across all three metrics (BLEU, ROUGE-L, chrF++), with InLegalTrans (FT) leading in all categories. The strong correlation between metrics validates the robustness of our evaluation. The model's strong performance in ROUGE-L (0.56) and chrF++ (0.73) metrics indicates robust sentence-level structural preservation and character-level accuracy.

4.2 Qualitative Analysis

Table 4 presents a representative translation example demonstrating the strengths and weaknesses of different models on legal text.

Table 4: Example translations from different models.

Model	Translation
Source	The petitioner has challenged the constitutional validity of Section 377.
Reference	याचिकाकर्ता ने धारा 377 की संवैधानिक वैधता को चुनौती दी है।
InLegalTrans (FT)	याचिकाकर्ता ने धारा 377 की संवैधानिक वैधता को चुनौती दी।
IndicTrans2 (FT)	याचिकाकर्ता ने अनुच्छेद 377 की संवैधानिक मान्यता को चुनौती दी।
NLLB-200	अर्जीदार ने सेक्शन 377 की संवैधानिक वैधता को चुनौती दी है।
Gemini (few-shot)	याचिकाकर्ता ने धारा 377 की वैधता पर सवाल उठाया है।

The qualitative analysis reveals that InLegalTrans (FT) produces translations nearly identical to the reference, maintaining precise legal terminology. IndicTrans2 substitutes वैधता (validity) with मान्यता (recognition), introducing a subtle but significant semantic shift in legal meaning. NLLB-200 uses inconsistent terminology (अर्जीदार instead of याचिकाकर्ता for petitioner) and transliterates "Section" as सेक्शन rather than using the proper Hindi term धारा. Gemini paraphrases excessively, changing "challenged" to "questioned" (सवाल उठाया), which alters the legal force and precision of the statement.

These examples confirm that domain-specific fine-tuning on legal corpora is essential for achieving high-quality English-Hindi legal translation, and that specialized models outperform general-purpose systems in preserving legal semantics.

5 Conclusion

In this paper, we present a comprehensive evaluation of neural machine translation models for the translation of English-Hindi legal documents as part of the L-MT shared task. We demonstrate that domain-specific fine-tuning on legal corpora substantially enhances translation quality for specialized legal texts, with the fine-tuned InLegalTrans model achieving the highest BLEU score of 0.48, a 55% improvement over its base performance.

Our comparative analysis of four translation systems spanning different architectural paradigms revealed that domain specialization and targeted fine-tuning can be more effective than raw model capacity, as evidenced by the 1B parameter InLegalTrans outperforming the larger 1.3B parameter NLLB-200 model. The results confirm that preserving legal terminology, statutory structure, and formal phrasing requires dedicated domain adaptation rather than relying solely on general-purpose multilingual models. As a natural extention of this work, we would explore the possibility of developing MT models from English to other Indian languages. Since the legal domain is a critical domain, it requires quality legal benchmarks to evaluate the developed models. We would like to work in this direction as well. We plan to introduce linguistic regularization mechanisms during training to explicitly model legal discourse markers and domain-specific cue phrases. The final fine-tuned model (InLegalTrans-FT) is available here - Hugging Face.

Limitations

While our system demonstrates strong performance on the L-MT dataset, several limitations warrant acknowledgment. The fine-tuning is performed exclusively on the provided legal corpus, which may limit generalization to other legal subdomains or regional legal language variations. The evaluation primarily relies on automatic metrics (BLEU, ROUGE-L, chrF++), which may not fully capture nuanced legal semantic equivalence. Although human evaluation at limited is carried out, it is conducted by non-experts. The system's handling of rare legal terminology and emerging legal concepts requires further extensive human evaluation by legal experts.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Deborah Cao. 2007. *Translating Law*. Multilingual Matters.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. Legal-bert: The muppets straight out of law school. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2898–2904.

Chenhui Chu and Rui Wang. 2018. A survey of domain adaptation for neural machine translation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1304–1319.

Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, and 1 others. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

M Amin Farajian, Marco Turchi, Matteo Negri, and Marcello Federico. 2017. Multi-domain neural machine translation through unsupervised adaptation. In *Proceedings of the Second Conference on Machine Translation*, pages 127–137.

Jay Gala, Pranjal A Chitale, AK Raghavan, Sumanth Doddapaneni, Varun Gumma, Aravinth Bheemaraj, Divyanshu Addanki, Divyanshu Kakwani, Anoop Kunchukuttan Sharma, Pratyush Kumar, and 1 others. 2023. Indictrans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages. In *Transactions on Machine Learning Research*.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Sayan Mahapatra, Debtanu Datta, Shubham Soni, Adrijit Goswami, and Saptarshi Ghosh. 2025. Milpac: A novel benchmark for evaluating translation of legal text to indian languages. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 24(8):1–30.

Vandan Mujadia, Ashok Urlana, Yash Bhaskar, Penumalla Aditya Pavani, Kukkapalli Shravya, Parameswari Krishnamurthy, and Dipti Sharma. 2024. Assessing translation capabilities of large language models involving English and Indian languages. In *Proceedings of the 25th Annual Conference of the European Association for Machine Translation (Volume 1)*, pages 207–228, Sheffield, UK. European Association for Machine Translation (EAMT).

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Maja Popović. 2017. chrf++: Words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation*, pages 612–618.

Gowtham Ramesh, Sumanth Doddapaneni, Aravinth Bheemaraj, Mayank Jobanputra, Raghavan AK, Ajitesh Sharma, Sujit Sahoo, Harshita Diddee, Mahalakshmi J, Divyanshu Kakwani, and 1 others. 2021. Samanantar: The largest publicly available parallel corpora collection for 11 indic languages. In *Transactions of the Association for Computational Linguistics*, volume 10, pages 145–162.

Carolina Rossi and Jean-Pierre Chevrot. 2019. Legal translation quality in the european union: A multi-method study. In *Proceedings of Machine Translation Summit XVII: Research Track*, pages 165–176.

Susan Šarčević. 2000. *New Approach to Legal Translation*. Kluwer Law International.

Kshetrimayum Boynao Singh, Deepak Kumar, and Asif Ekbal. 2025. Evaluation of llm for english to hindi legal domain machine translation systems. In *Proceedings of the Tenth Conference on Machine Translation (WMT 2025)*, pages 823–833. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mt5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498.

Haoxi Zheng, Mirella Lapata, and 1 others. 2021. Does pre-training on legal corpora improve legal language understanding? a case study on contract understanding. In *Proceedings of the Natural Legal Language Processing Workshop 2021*, pages 95–104.

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2024. Multilingual machine translation with large language models: Empirical results and analysis. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2765–2781, Mexico City, Mexico. Association for Computational Linguistics.

NIT-Surat@L-Summ: A Semantic Retrieval-Based Framework for Summarizing Indian Judicial Documents

Nita Jadav[1] **Ashok Urlana**[2] **Pruthwik Mishra**[1]
[1]SVNIT Surat, India
[2]TCS Research, Hyderabad, India
ds24ai001@aid.svnit.ac.in, ashok.urlana@tcs.com
pruthwikmishra@aid.svnit.ac.in

Abstract

The shared task of Legal Summarization (L-Summ) focuses on generating abstractive summaries for the Indian court judgments in English. This task presents unique challenges in producing fluent, relevant, and legally appropriate summaries given voluminous judgment texts. We experiment with different sequence-to-sequence models and present a comprehensive comparative study of their performance. We also evaluate various Large Language Models (LLM) with zero-shot settings for testing their summarization capabilities. Our best performing model is fine-tuned on a pre-trained legal summarization model where relevant passages are identified using the maximum marginal relevance(MMR) technique. Our findings highlight that retrieval-augmented fine-tuning is an effective approach for generating precise and concise legal summaries. We obtained a rank of 5th overall[1].

1 Introduction

One of the main impediments to India's growth is its pending cases, in which cases remain unsolved due to the shortage of competent manpower (Katju, 2019). According to the national judicial data grid[2], 82 percent of criminal cases are pending for more than 9 years in various district courts of India. The facts show that the Indian legal system alone has more than 44 million pending cases. Most of the processes in the legal system still remain manual with very few examples of transition to fully digital court rooms.

To assist the legal community, several works such as the series of shared tasks on Artificial Intelligence for Legal Assistance (AILA), semantic segmentation of judicial text using rhetorical roles (Malik et al., 2022), summarization (Parikha et al., 2021; Datta et al., 2023), machine translation (Kocmi et al., 2025), language understanding models specific to the Indian legal system (Paul et al., 2023), and legal reasoning models (Joshi et al., 2024) have been proposed. In a similar vein, the Legal Summarization shared task under the JustNLP workshop[3] attempts to develop legal summarization systems capable of handling domain-specific legal knowledge, legal understanding, legal reasoning, and coherent abstraction for Indian judgments in English.

Reasoning over large specialized legal judgments and generating abstractive summaries make this task even more challenging. As we are dealing with large documents, semantic segmentation of the document into passages becomes extremely important. Similarly, locating the relevant passages corresponding to a summary is also equally crucial for the overall performance of any summarization model especially in this domain.

Taking all these complexities into account, our contributions in this paper are threefold.

1. Semantic Segmentation of documents into meaningful passages using different LLMs
2. Relevant passage retrieval corresponding to a summary using different techniques with dense embeddings
3. Fine-tuning domain-specific summarization models with comprehensive comparison

2 Related Work

Although document summarization remains a very active and competitive research area in general and other technical domains, it has very sparse representation in the legal domain. Publicly available

[1]The demonstration of the approaches used in the paper can be found in https://github.com/nitajadav8/Legal_TextSumm
[2]https://njdg.ecourts.gov.in/njdg_v3/
[3]https://exploration-lab.github.io/JUST-NLP/

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 133–141

Metrics	Train		Validation		Test	
	Judgment	Summary	Judgment	Summary	Judgment	Summary
Number of Samples	1200	1200	200	200	400	400
(Min, Max) Words	(169, 149087)	(29, 2404)	(336, 92322)	(106, 1552)	(183, 155282)	(44, 1934)
Average Words	8294	606	6815	603	7920	621
(Min, Max) Sentences	(1, 3129)	(1, 18)	(2, 813)	(1, 13)	(1, 2228)	(1, 18)
Average Sentences	79	3	61	4	78	3
Compression Ratio	0.74		0.70		0.70	
Density	34.03		34.74		32.25	
Abstractivity	0.93		0.93		0.94	

Table 1: Dataset Statistics for Train, Validation, Test variants of L-Summ.

legal datasets are very few even for highly resource rich languages such as English. The majority of the available data span through sub-domains such as patents (Sharma et al., 2019) and contracts (Manor and Li, 2019), whereas actual court rulings (de Vargas Feijó and Moreira, 2018) are scarce.

To overcome the lack of rich resources, there have been few attempts in the Indian context to generate high-quality summarization datasets (Parikh et al., 2021). In-Abs (Shukla et al., 2022) consists of 7000 abstractive summaries or headnotes of the Indian Supreme Court judgments using reliable legal resources. IL-TUR (Joshi et al., 2024) presents a comprehensive benchmark and strong baseline models for the understanding and reasoning over the Indian legal texts. Their best performing summarization model is fine-tuned on the legal Pegasus (Sairam, 2021) model. The legal Pegasus model is a fine-tuned version of the Pegasus (Zhang et al., 2020) model on the US securities litigation dataset.

Most of the research in the domain of legal summarization is limited to only English. MILDSum (Datta et al., 2023) is an attempt to provide understandable summaries in native languages through cross-lingual summarization for complex court proceedings written in English. With the advent of LLMs and improved performance in the majority of tasks, the research in the direction of building domain-specific LLMs has gained traction. Domain adaptive pre-training and converting raw corpora into reading comprehension tasks (Cheng et al., 2023) enhance performance across various tasks in different domains, including law.

3 L-Sum Dataset Description

The dataset used in the task is shared by the organizers. Each entry in the dataset consists of a judgment and a corresponding concise summary in English. To perform in-depth data analysis, we adopted several metrics from Urlana et al. (2022, 2023b,a) and reported the number of words, the size of the vocabulary, compression-ratio and density of judgment to summary. The detailed data analysis statistics of the dataset are listed in Table 1.

4 Data processing

4.1 Preprocessing

The given documents in the L-Summ dataset do not have any annotations related to rhetorical roles (Malik et al., 2022) that could be leveraged for their semantic segmentation. Moreover, the document has additional noise of broken or incomplete sentences, misspelled words, non-ASCII characters, and title statements without proper structure. Considering this complexity, we incorporate different pre-processing techniques to remove the noise and divide each judgment into semantically coherent passages or paragraphs to harness the effectiveness of domain-specific summarization models that are explained in the following sections. All non-ASCII characters, special symbols, and redundant spaces are removed to ensure text uniformity.

4.2 Normalization

A judgment can have a wide variety of abbreviations from the legal domain as well as short sentences about sections, articles, writ numbers, and other legal information. It also contains unstructured sentences with improper spacing. Normalizing these sentences is essential for preparing the data for the models. We automate the normalization process using GPT-4 (OpenAI et al., 2024) and the corresponding prompt mentioned in Appendix A.1.

1. Expansion of Abbreviations: Common abbre-

	Model	R-2	R-L	BLEU
Zero-shot	Qwen-32b (Yang et al., 2025)	14.7	18.7	6.7
	gpt-oss-120b (Agarwal et al., 2025)	17.8	21.8	9.1
	Llama-4 17b (AI, 2025)	22.0	24.0	9.8
	Legal-Pegasus (Sairam, 2021)	17.0	20.7	6.2
	LongFormer (Beltagy et al., 2020)	21.6	21.9	15.4
	Llama 3.1 8b (Dubey et al., 2024)	21.1	23.8	10.5
	Llama 3 70b (Dubey et al., 2024)	21.4	23.6	7.9
Fine-tuning	Legal-Pegasus-DPR (Karpukhin et al., 2020)	19.4	20.2	9.9
	LongFormer-MCS (Shukla et al., 2022)	22.3	23.2	14.7
	Legal-Pegasus-MCS (Shukla et al., 2022)	22.9	23.0	9.6
	LongFormer-MMR (Xie and Liu, 2008)	21.7	21.9	15.6
	Legal-Pegasus-MMR (Xie and Liu, 2008)	**24.4**	**24.7**	**16.8**

Table 2: Various models' performance comparison on Validation Dataset.

viations such as *dept.*, *Addl.*, *secy.*, and *Lko* are expanded into their full forms—*department*, *Additional*, *Secretary* and *Lucknow*—to enhance text clarity and uniformity.

2. Coreference resolution of legal entities: Legal judgments often contain implicit or missing references to entities such as courts. In this step, such references are made explicit.

3. Completion of Incomplete Sentences: Fragmented or unstructured sentences are transformed into grammatically complete forms for better readability.

4.3 Passage Retrieval

Passage retrieval is very important in summarization tasks when we are dealing with large documents. It helps identify and extract relevant content that is semantically aligned with the summary, improving summary coherence and informativeness. We also perform experiments without considering the semantic relevance. In both experimental setups, the judgment document is divided into passages of a maximum of 1024 tokens, preserving the long sentence boundaries. The passage length is increased (maximum 2048 tokens) for models such as Longformer (Beltagy et al., 2020), as they are capable of handling large documents. For fine-tuning, three approaches are used to retrieve passages from a long judgment and its corresponding summary.

4.3.1 Dense Passage Retrieval (DPR)

The DPR approach (Karpukhin et al., 2020; Faizullah et al., 2024) is based on the idea of covering all summary sentences to identify their contributing passages from the judgment. The judgment is divided into passages with a maximum of 1024 tokens (to align with the model's capacity), covering whole sentences only. The cosine similarities between the embeddings of summary sentences and the embeddings of passages are calculated to find the most relevant passage for the summary sentence. We perform the experiment by identifying the three most relevant passages for each summary sentence. The training data is formed by combining all three relevant passages as the input and the corresponding summary sentence as the target.

Approach	R-2	R-L	BLEU
Legal-Pegasus-MCS	25.56	23.82	9.24
Legal-Pegasus-MMR	25.38	24.75	11.53
Legal-Pegasus-MMR-epoch:8	25.9	24.95	13.06

Table 3: Various methods performance comparison on Test Dataset.

4.3.2 Maximum Cosine Similarity (MCS)

In the MCS approach (Shukla et al., 2022), passages are ranked using maximum cosine similarity (MCS). Both documents and summaries are sentence-segmented and represented with SBERT embeddings (Reimers and Gurevych, 2019). For each document sentence, the most similar summary sentence is selected, and the process continues until reaching a 1024-token limit. The final training set thus contains passages capped by length and paired with their most relevant summary chunks.

4.3.3 Maximum Marginal Relevance (MMR)

To capture both semantic similarity and relevance between a judgment and its summary, we utilize the MMR (Xie and Liu, 2008; Zhong et al., 2019) approach. MMR balances two competing objectives: selecting passages that are *highly relevant* to the summary sentence while also maintaining *diversity* among the selected passages.

Let s denote the embedding of a summary sentence, and let $C = \{c_1, c_2, \ldots, c_n\}$ denote the set of embeddings of n candidate passages obtained from a judgment document. The relevance of each passage to the summary sentence is computed using the cosine similarity as follows:

$$\text{Rel}_i = \text{cosine\_sim}(s, c_i), \quad \forall i \in \{1, 2, \ldots, n\} \tag{1}$$

The pairwise similarity between two passages c_i and c_j is given by:

$$\text{Sim}_{ij} = \text{cosine\_sim}(c_i, c_j) \tag{2}$$

Step 1: Initial Selection. The first passage c_i to be selected is the one that is the most relevant to the summary sentence:

$$S_1 = \arg\max_i(\text{Rel}_i) \tag{3}$$

Step 2: Iterative Selection using MMR. For the remaining candidates, the MMR score is computed as:

$$\text{MMR\_score}(c_i) = \lambda \cdot \text{Rel}_i - (1-\lambda) \cdot \max_{c_j \in S} \text{Sim}_{ij} \quad (4)$$

where S is the set of passages already selected, and $\lambda \in (0,1)$ controls the trade-off between relevance and diversity. A higher λ value emphasizes *relevance* (passages similar to the summary sentence). A lower λ value emphasizes *diversity* (passages dissimilar to previously selected ones).

Step 3: Passage Update. The next passage to include is the one with the maximum MMR score:

$$S_i = \arg\max_{c_i \notin S} [\text{MMR\_score}(c_i)] \quad (5)$$

$$S \leftarrow S \cup \{S_i\} \quad (6)$$

Implementation Details. In our experiments, a maximum of two semantically relevant passages were selected for each summary sentence using $\lambda = 0.7$. A higher value of λ (> 0.5) ensures that retrieved passages are highly relevant to the summary sentence while maintaining moderate diversity.

Each selected passage is associated with its corresponding summary sentence in a mapping dictionary, since a passage may be relevant to multiple summary sentences. To retrieve semantically coherent passages, the judgment document is segmented into passages with a maximum token limit of 1024, ensuring complete sentences and a cosine similarity threshold above 0.2. For each retrieved passage, the corresponding summary chunk is then obtained using the constructed mapping dictionary.

5 Model Description

The summary generation from the preprocessed judgment is carried out using several popular and open-source models. All the models have demonstrated better efficiency in the general summarization task. The most accurate domain-specific models, like Legal-pegasus and Longformer, were also used to infer the result. The experiments were performed with zero-shot and fine-tuning models.

5.1 Zero-Shot Experiments

We utilize Qwen3 (Yang et al., 2025), gpt-oss (Agarwal et al., 2025), Llama 3 (Dubey et al., 2024), Llama 4 (AI, 2025), Longformer (Beltagy et al., 2020), and Legal-Pegasus (Sairam, 2021) to perform zero-shot experiments and obtain a summary from a preprocessed judgment. For LLM-based summary generation, Groq API[4] is used, which costs us a total of 26.32 dollars (USD) including normalization of raw judgments.

5.2 Fine-tuning Encoder-Decoder Models

We fine-tune two encoder-decoder (Sutskever et al., 2014) based models. The selection of models is based on the type of data on which those models were trained and the context length. The Longformer model is chosen because it can handle long documents similar to legal judgments. We select Legal-Pegasus as a base model, as it is fine-tuned on legal texts. The details of the hyper-parameters are given in Table 7. Fine-tuning is performed on the datasets created after passing the original documents through the pre-processing and normalization techniques, followed by the three passage retrieval techniques defined in Section 4.3.

6 Results and Evaluation

Each generated summary is evaluated on the basis of ROUGE-2, ROUGE-L (Lin, 2004) and BLEU (Papineni et al., 2002) scores. Table 2 and Table 3 present the results on the validation data and the test data, respectively. The approach of retrieving passages based on the MMR technique with the fine-tuned Legal-Pegasus performs the best. The efficiency of the MMR technique can be attributed to its ability to retrieve passages relevant to the summary with diversity among them. To improve the performance of the model, we also try different number of epochs for fine-tuning and the best performing model is fine-tuned for 8 epochs. The performance of LLMs to generate summaries from legal documents with zero-shot prompting is also compared. We also compare the results of different passage retrieval approaches. Table 9 shows the generated summaries for a sample document in Table 8 by models using different passage retrieval techniques.

Conclusion

As a part of the L-Summ shared task, we fine-tune domain-specific models using the semantic passage retrieval framework, considering the reference summary. Semantically relevant passage retrieval from

[4] https://console.groq.com/docs/models

large documents with domain knowledge plays a crucial role in improving the performance of summarization models. As a successor to this work, we are exploring techniques to fine-tune Legal LLMs.

7 Limitation

We analyze our results using the evaluation metrics recommended by the shared task organizers; however, the scope of this evaluation is limited and can be further expanded. Currently, our experiments are restricted to LLM-based zero-shot settings, and we have not performed any fine-tuning of LLMs due to computational constraints.

References

Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K Arora, Yu Bai, Bowen Baker, Haiming Bao, and 1 others. 2025. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*.

Meta AI. 2025. Llama 4 models.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *Preprint*, arXiv:2004.05150.

Daixuan Cheng, Shaohan Huang, and Furu Wei. 2023. Adapting large language models via reading comprehension. In *The Twelfth International Conference on Learning Representations*.

Debtanu Datta, Shubham Soni, Rajdeep Mukherjee, and Saptarshi Ghosh. 2023. MILDSum: A novel benchmark dataset for multilingual summarization of Indian legal case judgments. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5291–5302, Singapore. Association for Computational Linguistics.

Diego de Vargas Feijó and Viviane Pereira Moreira. 2018. Rulingbr: A summarization dataset for legal texts. In *International Conference on Computational Processing of the Portuguese Language*, pages 255–264. Springer.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407.

Abdur Rahman Bin Mohammed Faizullah, Ashok Urlana, and Rahul Mishra. 2024. Limgen: Probing the llms for generating suggestive limitations of research papers. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 106–124. Springer.

Abhinav Joshi, Shounak Paul, Akshat Sharma, Pawan Goyal, Saptarshi Ghosh, and Ashutosh Modi. 2024. Il-tur: Benchmark for indian legal text understanding and reasoning. *arXiv preprint arXiv:2407.05399*.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Justice Markandey Katju. 2019. Backlog of cases crippling judiciary.

Tom Kocmi, Ekaterina Artemova, Eleftherios Avramidis, Rachel Bawden, Ondrej Bojar, Konstantin Dranch, Anton Dvorkovich, Sergey Dukanov, Mark Fishel, Markus Freitag, and 1 others. 2025. Findings of the wmt25 general machine translation shared task: Time to stop evaluating on easy test sets. In *Proceedings of the Tenth Conference on Machine Translation, China. Association for Computational Linguistics*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Vijit Malik, Rishabh Sanjay, Shouvik Kumar Guha, Angshuman Hazarika, Shubham Nigam, Arnab Bhattacharya, and Ashutosh Modi. 2022. Semantic segmentation of legal documents via rhetorical roles. In *Proceedings of the Natural Legal Language Processing Workshop*, volume 2022, pages 153–171.

Laura Manor and Junyi Jessy Li. 2019. Plain English summarization of contracts. In *Proceedings of the Natural Legal Language Processing Workshop 2019*, pages 1–11, Minneapolis, Minnesota. Association for Computational Linguistics.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, and 262 others. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Vedant Parikh, Upal Bhattacharya, Parth Mehta, Ayan Bandyopadhyay, Paheli Bhattacharya, Kripabandhu Ghosh, Saptarshi Ghosh, Arindam Pal, Arnab Bhattacharya, and Prasenjit Majumder. 2021. Overview of the third shared task on artificial intelligence for legal assistance at fire 2021. In *Fire (working notes)*, pages 517–526.

Vedant Parikha, Vidit Mathurc, Parth Mehtab, Namita Mittalc, and Prasenjit Majumdera. 2021. Lawsum: A weakly supervised approach for indian legal document summarization. *arXiv preprint arXiv:2110.01188*.

Shounak Paul, Arpan Mandal, Pawan Goyal, and Saptarshi Ghosh. 2023. Pre-trained language models for the legal domain: A case study on indian law. In *Proceedings of 19th International Conference on Artificial Intelligence and Law - ICAIL 2023*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Naren Sairam. 2021. Legal pegasus model.

Eva Sharma, Chen Li, and Lu Wang. 2019. BIGPATENT: A large-scale dataset for abstractive and coherent summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2204–2213, Florence, Italy. Association for Computational Linguistics.

Abhay Shukla, Paheli Bhattacharya, Soham Poddar, Rajdeep Mukherjee, Kripabandhu Ghosh, Pawan Goyal, and Saptarshi Ghosh. 2022. Legal case document summarization: Extractive and abstractive methods and their evaluation. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1048–1064, Online only. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27.

Ashok Urlana, Sahil Manoj Bhatt, Nirmal Surange, and Manish Shrivastava. 2023a. Indian language summarization using pretrained sequence-to-sequence models. In *Fire*.

Ashok Urlana, Pinzhen Chen, Zheng Zhao, Shay Cohen, Manish Shrivastava, and Barry Haddow. 2023b. PMIndiaSum: Multilingual and cross-lingual headline summarization for languages in India. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11606–11628, Singapore. Association for Computational Linguistics.

Ashok Urlana, Nirmal Surange, Pavan Baswani, Priyanka Ravva, and Manish Shrivastava. 2022. TeSum: Human-generated abstractive summarization corpus for Telugu. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 5712–5722, Marseille, France. European Language Resources Association.

Shasha Xie and Yang Liu. 2008. Using corpus and knowledge-based similarity measure in maximum marginal relevance for meeting summarization. In *2008 IEEE international conference on acoustics, speech and signal processing*, pages 4985–4988. IEEE.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 11328–11339. PMLR.

Linwu Zhong, Ziyi Zhong, Zinian Zhao, Siyuan Wang, Kevin D. Ashley, and Matthias Grabmair. 2019. Automatic summarization of legal decisions using iterative masking of predictive sentences. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Law*, ICAIL '19, page 163–172, New York, NY, USA. Association for Computing Machinery.

A Appendix

A.1 Examples of Prompt

For the data normalization task, the prompt consists of instructions to expand abbreviations, standardize entity names, structuring broken sentences. We design prompt in such a way that it covers all requirements for normalizing judgment which can be used later for semantic passage retrieval. Here, we instruct the LLM to preserve the original meaning of sentences without paraphrasing and also avoid summarizing at this phase. Table 6 gives the prompt sample, we used for performing the normalization and generation parameter values in our experiment. The sample outcome of normalization is given in Table 4.

We design summarization prompts to generate an abstractive summary with clear instruction to use domain specific and fluent legal terms. We also constraint the summary length maximum of 500 tokens as per the rules of shared task. Our prompt example is given in Table 5 with generation parameters.

The LLM based experiments are performed using the Groq API [5] for high speed inference. The Groq API offers a free-tier usage limit of tokens per day per api call for selected large language models (LLM). To adhere to this constraint, the judgment texts were divided into smaller passages, ensuring that each segment remained within the maximum token limit which in our case was 50000 tokens. All the Groq inferences conducted by setting the temperature=0 to encourage the reproducibility. Considering the shared task deadline and the need for higher inference capacity, the Groq API key was subsequently upgraded to the developer tier to support extended model utilization.

Raw text: Case :- WRIT - C No. - 11383 of 2023 Petitioner :- Syed Hamidul Bari Respondent :- State Of U.P. Thru. Addl. Chief/Prin. Secy. Housing And Urban Planning Deptt. Lko. And 4 Others
Normalized text: Case: Writ Petition No. 11383 of 2023, Petitioner: Syed Hamidul Bari, Respondent: State of Uttar Pradesh through Additional Chief/Principal Secretary, Housing and Urban Planning Department, Lucknow, and four others.

Table 4: Example of the Raw text Vs. normalized text sample.

[5] https://console.groq.com

You are an abstractive text summarizer for legal document in the Indian legal domain. Clean the document by removing unwanted noisy data and frame sentences in clear legal-style English. Produce a single fluent abstractive summary containing Context and background of the case with the court name, Legal reasoning and Final decision/outcome made by court. Return the result as a summarized paragraph in human-readable form with a maximum of 500 words. Document:**{doc}** **End-of-Prompt**

Table 5: Prompt Template for zero-shot prompting.

A.2 Fine Tuning Parameters

In the fine-tuning phase, several key hyperparameters were configured to optimize model performance while maintaining computational efficiency. To address GPU memory constraints encountered during the fine-tuning of long-former models, the gradient accumulation steps were set to 4, allowing effective larger batch sizes without exceeding memory limits. For effective training, we consider 20% of train data as validation data. To increase our score on the best performing model, we finally fine-tune the legal pegasus model with eight epochs to infer the test results. All experiments are carried out on a single NVIDIA H-100 GPU with 94GB RAM. The complete set of selected hyperparameter values is summarized in Table 7.

Parameter Name	Legal-Pegasus	LongFormer
Epochs	4, 8	4
Train Batch	1	1
Evaluation Batch	1	1
Warmup Step	200	750
Gradient accumulation steps	1	4
Evaluation accumulation steps	1	1
Learning Rate	1e-4	1e-4

Table 7: Hyperparameters for Fine-tuning

Normalizing Prompt:

You are a text cleaning expert for Indian legal case documents. Take the following legal document text and split it into clean paragraphs based on following conditions.

1. Remove meaningless statements, 2. Expand abbreviations according to the Indian legal context, 3. Use full court names according to the Indian legal system instead of short forms, 4. Add full court name if not present, 5. Remove empty lines, 6. Remove numbering like "1.", "(i)", "(a)" from the beginning of lines, 7. Merge broken lines into proper sentences, 8. Remove Unicode or non-standard characters, 9. Do **not** rephrase, 10. Do **not** summarize, 11. Correct spelling of misspelled or noisy English words containing digits or non-alphabetic characters, 12. Form paragraphs containing at most 1000 tokens, keeping complete sentences without breaking them, 13. Return the output **only** as a JSON array of paragraphs.

Document: `{doc}`

End-of-Prompt

Table 6: Prompt template for GPT-4-based normalization of Indian legal documents.

Judgment: Petitioner :- Suresh Devi And Another Respondent :- State Of U P And 13 Others Counsel for Petitioner :- Jamil Ahamad Azmi,Anand Swaroop Gautam,Ashutosh Kumar Tiwari,Dharmendra Singh Counsel for Respondent :- G A Hon'ble Ashwani Kumar Mishra,J Hon'ble Deepak Verma,J This petition has been filed invoking jurisdiction of this Court under Article 226 of the Constitution of India with the allegation that petitioner's son has been murdered in police custody in the night intervening 11/12th December, 2020 while he was in police custody in Case Crime No 1181 of 2020, under Section 366 I P C , Police Station Khurja Nagar, District Bulandshahar A prayer has also been made to protect the life and liberty of the petitioners as their son was subjected to torture and murder as he had contracted inter caste marriage out of his own free will It is contended that the authorities have been most unfair in dealing with the grievance raised by the petitioners Attention of the Court has been invited to the provision contained in Section 176 Cr P C , sub Section (1)A whereof contemplates holding of a judicial enquiry where death is caused in the custody of the police It is also stated that neither any post mortem has been carried out nor the body has been burried and instead the police personnels have cremated the body contrary to all settled norms Learned AGA does not dispute the fact that a judicial enquiry was initiated with a request made to District Judge on 06 01 2021 and states that a report is still awaited In matters, where the allegation is with regard to custodial death, judicial enquiry under Section 176 (1)A cannot be allowed to drag for so long These are instances which have to be viewed with greatest sensitivity and concern We, therefore, direct the Registry to enquire from the District Judge, Bulandshahar as to when the enquiry report has been submitted in the matter and, in the event, such a report is not submitted, the explanation of the Judicial Officer in that regard shall be placed before us by the next date fixed as a period of more than one year has expired We hasten to add that in the event, such enquiry has not been concluded so far, the same shall be concluded most expeditiously by following the procedure in law List this matter, once again, on 27 01 2022 at 02:00 PM

Reference Summary: The Allahabad High Court recently observed that judicial inquiry in custodial death cases cannot be dragged on for long [Suresh Devi and Another v. State of Uttar Pradesh and Others]. A Division Bench of Justices Ashwani Kumar Mishra and Deepak Verma held, "In matters, where the allegation is with regard to custodial death, judicial enquiry under Section 176 (1)A cannot be allowed to drag for so long. These are instances which have to be viewed with greatest sensitivity and concern." The Court was hearing a plea alleging that the petitioner's son was murdered in police custody in December 2020. It was claimed that the son was tortured and killed while in police custody after he had contracted inter-caste marriage out of his own free will. The petitioner had also prayed for a direction to protect the life and liberty of the parents of the deceased. It was the case of the petitioner that the concerned authorities have been most unfair in dealing with their grievances, as no post-mortem was carried out on the body of the deceased, which was cremated contrary to all settled norms. Moreover, the Court's attention was drawn to the judicial enquiry initiated by the District Judge, Bulandshahr, on January 6, 2021. A report on the same was still awaited. While noting that more than one year has expired since the judicial enquiry was ordered, the Bench directed the High Court Registry to enquire from the District Judge, Bulandshahr as to when the enquiry report was submitted. In the event such a report is not submitted, the explanation of the judicial officer was directed to be placed before the Court. "We hasten to add that in the event, such enquiry has not been concluded so far, the same shall be concluded most expeditiously by following the procedure in law," the Court added. The matter was listed on January 27 for further hearing. Advocates Jamil Ahamad Azmi, Anand Swaroop Gautam, Ashutosh Kumar Tiwari and Dharmendra Singh appeared for the petitioners.

Table 8: Document-Summary Sample

Legal-Pegasus-MCS: In matters, where the allegation is with regard to custodial death, judicial enquiry under Section 176 (1)A cannot be allowed to drag for so long.These are instances which have to be viewed with greatest sensitivity and concern.We, therefore, direct the Registry to enquire from the District Judge, Bulandshahar as to when the enquiry report has been submitted in the matter and, in the event, such a report is not submitted, the explanation of the Judicial Officer in that regard shall be placed before us by the next date fixed as a period of more than one year has expired.We hasten to add that in the event, such enquiry has not been concluded so far, the same shall be concluded most expeditiously by following the procedure in law.List this matter, once again, on 27.01.2022 at 02:00 PM.We, therefore, direct the Registry to enquire from the District Judge, Bulandshahar as to when the enquiry report has been submitted in the matter and, in the event, such a report is not submitted, the explanation of the Judicial Officer in that regard shall be placed before us.
Legal-Pegasus-DPR: A prayer has also been made to protect the life and liberty of the petitioners as their son was subjected to torture and murder as he had contracted inter-caste marriage out of his own free will. We hasten to add that in the event, such enquiry has not been concluded so far, the same shall be concluded most expeditiously by following the procedure in law.
Legal-Pegasus-MMR: Advocates Jamil Ahamad Azmi, Anand Swaroop Gautam, Ashutosh Kumar Tiwari, and Dhirajendra Singh appeared for the petitioner. Advocate Ashwani Kumar Mishra and Justice Deepak Verma represented the State of UP. The Court was hearing a petition filed by the father of a man who was found dead in police custody on December 11, 2020 while he was in custody in a case related to his alleged inter-caste marriage out of his own free will. Ä prayer has also been made to protect the life and liberty of the petitioners as their son was subjected to torture and murder as he had contracted inter-caste marriage out of his own free will. It is contended that the authorities have been most unfair in dealing with the grievance raised by the petitioners,ẗhe Court said. The Allahabad High Court recently observed that judicial inquiry cannot be allowed to drag on for so long when it comes to custodial death [Suresh Devi v. State of UP]. In matters where the allegation is with regard to custodial death, judicial enquiry under Section 176 (1)A cannot be allowed to drag for so long. These are instances which have to be viewed with greatest sensitivity and concern. Ẅe, therefore, direct the Registry to enquire from the District Judge, Bulandshahar as to when the enquiry report has been submitted in the matter and, in the event, such a report is not submitted, the explanation of the Judicial Officer in that regard shall be placed before us by the next date fixed as a period of more than one year has expired,ẗhe Court ordered. The Allahabad High Court on Wednesday directed the registry to enquire from the District Judge, Bulandshahar as to when the enquiry report has been submitted in the matter and, in the event, such a report is not submitted, the explanation of the Judicial Officer in that regard shall be placed before us by the next date fixed as a period of more than one year has expired.

Table 9: Generated Summaries for the Judgment in Table 8

Adapting IndicTrans2 for Legal Domain MT via QLoRA Fine-Tuning at JUST-NLP 2025

Akoijam Jenil Singh[1], **Loitongbam Sanayai Meetei**[2], and **Yumnam Surajkanta**[1]

[1]NIT Manipur, Manipur, India
[2]SOA University, Odisha, India
{akoijamjenilsingh, loisanayai, ysurajkanta}@gmail.com

Abstract

Machine Translation (MT) in the legal domain presents substantial challenges due to its complex terminology, lengthy statutes, and rigid syntactic structures. The JUST-NLP 2025 Shared Task on Legal Machine Translation [1] was organized to advance research on domain-specific MT systems for legal texts. In this work, we propose a fine-tuned version of the pretrained large language model (LLM) ai4bharat/indictrans2-en-indic-1B [2], a transformer-based English-to-Indic translation model. Fine-tuning was performed using the parallel corpus provided by the JUST-NLP 2025 Shared Task organizers.Our adapted model demonstrates notable improvements over the baseline system, particularly in handling domain-specific legal terminology and complex syntactic constructions. In automatic evaluation, our system obtained BLEU = 46.67 and chrF = 70.03.In human evaluation, it achieved adequacy = 4.085 and fluency = 4.006. Our approach achieved an AutoRank score of 58.79, highlighting the effectiveness of domain adaptation through fine-tuning for legal machine translation. [3]

1 Introduction

India is a linguistically diverse country, with 22 officially recognized languages listed under the Eighth Schedule of the Constitution as of 2004. Despite this multilingual landscape, English serves as the official language of the judiciary throughout the country. In certain states such as Rajasthan, Madhya Pradesh, Uttar Pradesh, and Bihar, the use of Hindi is also permitted in High Court proceedings (PBI, 2025), highlighting the need for high-quality legal translation systems between English and Hindi.

However, legal translation is uniquely complex due to the presence of domain-specific terminology, lengthy statutes, and highly formalized language structures. General-purpose machine translation systems are not designed to handle such intricacies. Even minor translation errors in legal contexts can result in significant misunderstandings, making precision and domain awareness critical requirements.

The JUST-NLP 2025 Shared Task aims to advance machine translation in the legal domain, focusing on the English–Hindi language pair. In this paper, we present a domain-adapted legal machine translation system built upon the pretrained indictrans2-en-indic-1B model (Gala et al., 2023). The pretrained was originally developed for general-purpose translation across the 22 languages listed in the Eighth Schedule of the Indian Constitution. We fine-tune the model on the legal parallel corpus provided by the JUST-NLP 2025 Shared Task. Fine-tuning on this domain-specific corpus enhances the system's robustness in translating legal texts from English to Hindi, ensuring better preservation of legal terminology and contextual accuracy.

As part of the JUST-NLP 2025 Shared Task on Legal Machine Translation, our system demonstrated strong performance, achieving an AutoRank score of 58.79. This result provides empirical evidence that domain adaptation substantially enhances translation quality in the legal domain.

To support reproducibility and facilitate further research, we release the fine-tuned weights of our model, built on top of indictrans2-en-indic-1B. The model weights are publicly available at our repository [4].

[1]JUST-NLP 2025 Shared Task.
[2]Hugging-Face ai4bharat/indicTrans2-en-indic-1B.
[3]The final result announced by JUST-NLP 2025 Shared Task organizers
[4]Repository of the Model Weight

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 142–147
December 24, 2025

2 Related Work

Machine Translation (MT) is a core task in Natural Language Processing (NLP), aiming to automatically translate text across languages. In Indian language and legal translation, Haque et al. (2019) applied Phrase-Based SMT for English–Hindi, and Das et al. (2025) extended SMT to fifteen Indic languages. Evaluation of English–Hindi systems by Shetty (2025) found Google Translate and IndicTrans2 to achieve the highest automatic scores. More recently, Singh et al. (2025) assessed thirty-seven LLMs for English-to-Hindi legal translation, identifying Gemini-2.5-Pro, ONLINE-B, and Claude-4 as top performers. The MultiIndic22MT 2024 shared-task (Singh et al., 2024) focused on English–Manipuri translation using Transformer-based NMT with OpenNMT, comparing sequence-to-sequence models and Byte Pair Encoding (BPE) tokenization.

Overall, research shows a shift from rule-based and phrase-based methods to neural and transformer architectures. Nevertheless, accurate legal translation between English and Hindi remains challenging due to limited domain-specific corpora, complex terminology, and contextual ambiguities. The next section addresses these challenges using transformer-based architectures combined with domain adaptation techniques for legal machine translation.

3 Dataset

The JUST-NLP 2025 Shared Task focuses on translating legal texts from English (source) to Hindi (target). The organizers provided three Excel files: `English-hindi-train.xlsx` (tra, 2025), `English-hindi-valid.xlsx` (val, 2025), and `WMT25-TS_eng-hin-test.xlsx` (tes, 2025). The training file contains 50,000 English–Hindi parallel sentence pairs from the legal domain, while the validation and test files each contain 5,000 English-only sentences for evaluation and testing, respectively.

To facilitate model training and hyperparameter tuning, we further split the training data into 48,000 sentence pairs for training and 2,000 for internal validation. The official test set (tes, 2025) is used for final evaluation of our system using automatic metrics such as BLEU, chrF, and METEOR. Table 1 summarizes the dataset structure.

Dataset	**Size (pairs)**
`Train (full)`	50,000
`Train (used)`	48,000
`Validation (used)`	2,000
Dataset	**Source Only**
`Validation (official)`	5,000
`Test (official)`	5,000

Table 1: JUST-NLP 2025 dataset split statistics for English–Hindi legal text translation.

4 Methodology

We propose an English-to-Hindi machine translation system tailored for the legal domain. Our approach builds upon the pretrained IndicBART model indictrans2-en-indic-1B which we fine-tune using a domain-specific parallel corpus. This fine-tuning process allows the model to more effectively learn and translate legal terminology and contextual nuances, resulting in translations that are both accurate and contextually appropriate for legal texts. Following the fine-tuning, a post-processing step is applied to remove any unwanted characters produced during translation. A visual overview of this process is provided in Figure 1.

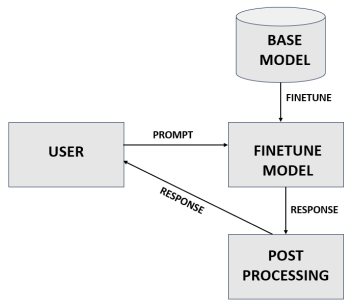


Figure 1: Workflow of the English→Hindi legal machine translation system, including user prompt, fine-tuning the base model, and post processing.

4.1 Data Preprocessing

We prepare our dataset by loading and tokenizing all the training, validation and test corpora using the sequence-to-sequence tokenizer provided by the base model indictrans2-en-indic-1B. This ensures consistency with the input format expected by IndicBART and preserves syntactic and semantic structures necessary for high-quality translation.

4.2 Parameter-Efficient Fine-Tuning via QLoRA

Due to computational constraints, we adopt QLoRA (Quantized Low-Rank Adapter) (Dettmers et al., 2023) for efficient fine-tuning. QLoRA combines 4-bit quantization of the pretrained model with Low-Rank Adaptation (LoRA), which introduces trainable adapter layers into specific transformer components while keeping the base model weights frozen. This method significantly reduces GPU memory usage and training cost, enabling fine-tuning of large language models (LLMs) without substantial degradation in performance.

Parameter	Setting
4 bit quantization	True
Device map	auto
LoRA rank (r)	16
LoRA alpha	16
LoRA dropout	0.05
Task type	Seq2Seq LM

Table 2: Key Hyperparameters for QLoRA-based Fine-Tuning.

4.3 Training Strategy

We fine-tuned the model indictrans2-en-indic-1B using 4-bit QLoRA (Quantized Low-Rank Adaptation) from the Hugging Face library (Hug, 2025) for parameter-efficient training.

Training Args	Values
Optimizer	AdamW
Learning Rate	2e-4
Scheduler	Cosine Scheduler
Weight Decay	0.01
GPU	NVIDIA T4
Batch Size	16
Mixed Precision	fp16
Checkpoint	Every 1000 steps
Tokenizer	Seq2Seq Tokenizer
Dynamic Padding	DataCollatorForSeq2Seq

Table 3: Training parameters for finetuning ai4bharat/indictrans2-en-indic-1B.

We train the model using an early stopping mechanism with a patience value of 5. The three best-performing checkpoints are selected based on validation loss. These checkpoints are then ensembled to form the final model, aggregating outputs to improve robustness and translation quality.

We apply a post-processing step to clean the model outputs. Specifically, we remove extraneous characters, such as punctuation marks, which are occasionally generated at the end of translated sentences. This step helps improve the fluency and readability of the final output and ensures conformity with the target language conventions.

4.4 Inference

During inference, the fine-tuned model is loaded, and source sentences are tokenized accordingly. Target sequences are generated using beam search with a beam width of 5 and a maximum length of 512 tokens. We evaluated multiple beam widths and observed that this setting yields the best translation performance.

5 Experiments and Results

We fine-tuned the pretrained indictrans2-en-indic-1B model on the English–Hindi legal parallel corpus to adapt it to the legal domain.

5.1 Evaluation Metrics

We conducted a human evaluation focusing on adequacy and fluency. In addition, the translations produced by our model were evaluated by the shared task organizers using automatic metrics. The evaluation procedures are described below.

5.1.1 Human Evaluation Metrics

Human evaluation remains the most reliable approach for assessing translation quality, as it captures linguistic and semantic nuances that automatic metrics may overlook. We conducted human evaluation along two qualitative dimensions: adequacy and fluency. These metrics provide complementary insights into translation performance and are described below.

Adequacy Evaluation : Adequacy (Snover et al., 2009) measures the extent to which the translated text preserves the meaning of the source sentence, regardless of its grammatical quality.

Fluency Evaluation : Fluency (Snover et al., 2009) assesses the grammatical correctness and naturalness of the translation in the target language, independent of the source content.

The scoring criteria of Adequacy and Fluency Evaluation is given in table 4

Score	Adequacy	Fluency
1	Does not retain any of the information from the source sentence.	Unintelligible due to grammatical errors.
2	Conveys only a minimal amount of information.	Contains grammatical errors that impede comprehension.
3	Retains a moderate amount of information.	Includes some mistakes or phrasing that feels unnatural.
4	Retains almost all relevant information.	Conforms to accepted grammatical norms.
5	Accurately reflects all information in the source.	Flawless, natural, and stylistically appropriate.

Table 4: Human evaluation criteria for fluency and adequacy. Reprodced from (Meetei et al., 2024)

5.1.2 Automatic Evaluation Metrics

Automatic evaluation is widely adopted in machine translation research for its scalability, reproducibility, and efficiency. The automatic metrics employed in this work are described below.

BLEU : The Bilingual Evaluation Understudy (BLEU) score (Papineni et al., 2002) measures the n-gram precision of a candidate translation with respect to reference translations, penalizing short translations with a brevity penalty.

ChrF : The Character F-score (ChrF) (Popović, 2015) calculates F-scores over character n-grams rather than word n-grams, which makes it more suitable for morphologically rich languages.

METEOR : METEOR (Metric for Evaluation of Translation with Explicit ORdering) (Banerjee and Lavie, 2005) aligns hypothesis and reference sentences based on exact, stem, synonym and paraphrase matches. A higher METEOR score reflects better adequacy and fluency.

TER : Translation Edit Rate (TER) (Snover et al., 2006) measures the number of edits required to transform the system output into the reference translation. Lower TER values indicate higher translation quality, as fewer edits are needed to match the human reference.

BERTScore : BERTScore (Zhang et al., 2019)leverages contextual embeddings from pre-trained language models to compute semantic similarity between hypothesis and reference. Higher scores indicate a stronger semantic alignment.

COMET : COMET (Crosslingual Optimized Metric for Evaluation of Translation) (Rei et al., 2020) is a neural evaluation metric trained to predict human judgments of translation quality. Higher COMET scores indicate closer agreement with human assessments of adequacy and fluency.

5.2 Results

The performance of our fine-tuned English→Hindi legal MT system is summarized through human evaluation and official leaderboard results from the JUST-NLP 2025 Shared Task.

Human evaluation was conducted by bilingual experts fluent in English and Hindi. Adequacy and fluency scores are reported in Table 5. The results indicate strong preservation of meaning and natural readability in Hindi translations.

Model	Adequacy	Fluency
Finetuned Model	4.085	4.006

Table 5: Human evaluation of the English→Hindi legal MT system. Scores range from 1 (poor) to 5 (excellent).

On the official leaderboard, our system achieved strong n-gram overlap, morphological robustness, and semantic preservation: BLEU = 46.67, METEOR = 72.86, TER = 44.63, chrF++ = 70.03, BERTScore = 90.86, and COMET = 72.12. The AutoRank score, computed by the organizers as a weighted combination of these metrics, is 58.79, indicating high-quality translations. The AutoRank calculation is given in Equation 1.

Leaderboard Results. Table 6 presents the top 7 participants for English→Hindi legal translation. Metrics include BLEU, METEOR, TER, chrF++, BERTScore, COMET, and AutoRank. Our system, **JUST-MEI**, ranked 5th, demonstrating competitive performance across all metrics.

Overall, both automatic and human evaluations confirm that our QLoRA fine-tuned IndicTrans2 model reliably translates English legal texts into Hindi, maintaining high lexical, semantic, and stylistic accuracy while effectively preserving legal terminology.

$$\text{AutoRank} = \frac{1}{6}\Big(BLEU_{\text{norm}} + METEOR_{\text{norm}} + (1 - TER_{\text{norm}}) + CHRF^{++}_{\text{norm}} + BERTScore_{\text{norm}} + COMET_{\text{norm}}\Big) \quad (1)$$

Rank	Team	BLEU↑	METEOR↑	TER↓	chrF++↑	BERTScore↑	COMET↑	AutoRank↑
1	Team-SVNIT	51.61	75.80	37.09	73.29	92.61	76.36	61.62
2	FourCorners	50.19	69.54	42.32	73.67	92.70	75.74	60.31
3	goodmen	48.56	67.15	41.63	73.07	92.38	75.16	59.39
4	JUNLP	46.03	71.84	42.08	70.59	91.19	73.72	58.90
5	**JUST-MEI**	**46.67**	**72.86**	**44.63**	**70.03**	**90.86**	**72.12**	**58.79**
6	Lawgorithms	46.27	71.80	43.06	68.32	91.03	72.14	58.26
7	Tokenizers	34.08	61.78	55.25	56.75	87.93	65.20	50.87

Table 6: Top 7 participants in the JUST-NLP 2025 Shared Task for English→Hindi legal translation. Automatic metrics reflect both formal correctness and semantic accuracy. Our system (rank 5) is highlighted in bold.

English (Source)	Hindi (Finetuned Model)	Legal Term Correctness
plaintiff No.1 was dead.	वादी संख्या 1 की मृत्यु हो चुकी थी ।	correct
hence, this appeal.	अतएव, यह अपील	correct
writ petition is dismissed.	रिट याचिका खारिज की जाती है ।	correct
they were employees employed under the defendant-appellants.	वे प्रतिवादीगण – अपीलार्थीगण के अधीनियोजित कर्मचारीगण थे ।	correct
other allegations were denied by the defendant.	प्रतिवादी द्वारा अन्य अभिकथनों से इनकार किया गया था ।	correct
accordingly, the title appeal was dismissed.	तद्नुसार, अभिधान अपील खारिज कर दी गयी थी ।	Legal term correct; minor lexical mismatch
PW-36 is the plaintiff himself.	अ जनतादल – 36 स्वयं वादी है ।	Partially correct; witness designation mistranslated

Table 7: Sample English→Hindi legal translations showing preservation of legal terminology. Each entry is evaluated for correctness of domain-specific terms.

5.3 Preservation of Legal Terminology

We evaluated whether the translations correctly preserve the legal terminology. Most legal terms were accurately rendered in Hindi, reflecting the model's ability to capture domain-specific terminology. However, a small portion of terms were mistranslated or rendered in a non-standard form, indicating that while the system is largely effective in maintaining legal terminology, occasional inconsistencies remain. Table 7 shows the sample output.

6 Conclusion and Future Work

We presented a domain-adapted English-to-Hindi legal machine translation system built on the pre-trained indictrans2-en-indic-1B model and fine-tuned with QLoRA on the JUST-NLP 2025 legal corpus. Our approach effectively captures domain-specific terminology and contextual nuances, yielding substantial improvements over a general-purpose baseline across multiple automatic metrics (BLEU, METEOR, TER, chrF++, BERTScore, COMET) and human evaluation dimensions (adequacy and fluency). The results demonstrate that the proposed system produces accurate and natural translations, highlighting the effectiveness of domain adaptation and the importance of combining automatic and human evaluations for comprehensive evaluation in specialized translation settings such as the legal domain.

Although our study is limited to a single model variant and limited computational resources, future work can investigate larger architectures, multilingual legal translation, and advanced domain adaptation techniques to further enhance performance. In general, our results highlight the importance of targeting domain adaptation for producing accurate and reliable legal machine translation systems in the Indian context.

Limitation

Although our fine-tuned model demonstrates strong performance, several limitations remain. First, we only explored a single variant of Indic-

Trans2; other architectures and larger models were not evaluated. Additionally, our experiments were constrained by hardware limitations, including limited GPU resources and batch sizes. To accommodate these constraints during fine-tuning, we employed 4-bit quantization of the base model.

Acknowledgments

We thank the organizers of the JUST NLP 2025 Shared Task for providing the dataset and the competition platform. We also acknowledge the support and valuable feedback from our colleagues and team.

References

2025. Huggin face peft docs. https://huggingface.co/docs/peft/index. Accessed: 2025-10-25.

2025. testing dataset. https://huggingface.co/datasets/helloboyn/WMT25-TS/tree/main. Accessed: 2025-10-25.

2025. training dataset. https://huggingface.co/datasets/helloboyn/IJCNLP-JustNLP-LMT/blob/main/english-hindi-train.xlsx. Accessed: 2025-10-25.

2025. Using regional language in court. https://www.pib.gov.in/Pressreleaseshare.aspx?PRID=2042983. Accessed: 2025-10-24.

2025. validation dataset. https://huggingface.co/datasets/helloboyn/IJCNLP-JustNLP-LMT/resolve/main/english-hindi-valid.xlsx. Accessed: 2025-10-25.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Sudhansu Bala Das, Divyajyoti Panda, Tapas Kumar Mishra, and Bidyut Kr Patra. 2025. Statistical machine translation for indic languages. *Natural Language Processing*, 31(2):328–345.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. *Advances in neural information processing systems*, 36:10088–10115.

Jay Gala, Pranjal A Chitale, Raghavan AK, Varun Gumma, Sumanth Doddapaneni, Aswanth Kumar, Janki Nawale, Anupama Sujatha, Ratish Puduppully, Vivek Raghavan, and 1 others. 2023. Indictrans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages. *arXiv preprint arXiv:2305.16307*.

Rejwanul Haque, Md Hasanuzzaman, and Andy Way. 2019. Investigating terminology translation in statistical and neural machine translation: a case study on english-to-hindi and hindi-to-english. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 437–446.

Loitongbam Sanayai Meetei, Thoudam Doren Singh, and Sivaji Bandyopadhyay. 2024. An empirical study of a novel multimodal dataset for low-resource machine translation. *Knowledge and Information Systems*, 66(11):7031–7055.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Maja Popović. 2015. chrf: character n-gram f-score for automatic mt evaluation. In *Proceedings of the tenth workshop on statistical machine translation*, pages 392–395.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. Comet: A neural framework for mt evaluation. *arXiv preprint arXiv:2009.09025*.

Ahan Prasannakumar Shetty. 2025. Evaluating machine translation models for english-hindi language pairs: A comparative analysis. *arXiv preprint arXiv:2505.19604*.

Kshetrimayum Boynao Singh, Deepak Kumar, and Asif Ekbal. 2025. Evaluation of llm for english to hindi legal domain machine translation systems. In *Proceedings of the Tenth Conference on Machine Translation, China. Association for Computational Linguistics*.

Ningthoujam Justwant Singh, Kshetrimayum Boynao Singh, Ningthoujam Avichandra Singh, Sanjita Phijam, and Thoudam Doren Singh. 2024. Wmt24 system description for the multiindic22mt shared task on manipuri language. In *Proceedings of the Ninth Conference on Machine Translation*, pages 797–803.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 223–231.

Matthew Snover, Nitin Madnani, Bonnie Dorr, and Richard Schwartz. 2009. Fluency, adequacy, or hter? exploring different human judgments with a tunable mt metric. In *Proceedings of the fourth workshop on statistical machine translation*, pages 259–268.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Team-SVNIT at JUST-NLP 2025: Domain-Adaptive Fine-Tuning of Multilingual Models for English–Hindi Legal Machine Translation

Rupesh Dhakad and **Naveen Kumar** and **Shrikant Malviya**
Sardar Vallabhbhai National Institute of Technology, Surat
dhakadrupesh666@gmail.com,{naveenkumar,shrikant}@coed.svnit.ac.in

Abstract

Translating sentences between English and Hindi is challenging, especially in the domain of legal documents, due to the specialized legal terminology and the lengthy, complex sentences that often accompany them. In this paper, we fine-tune and compare multiple pretrained multilingual translation models, including the `facebook/nllb-200-distilled-1.3B`, on a corpus of 50,000 English–Hindi legal sentence pairs provided for the shared task. The training pipeline includes preprocessing, context windows of 512 tokens, and decoding methods to enhance translation quality. The proposed method secured **1st place** on the official leaderboard. We obtained the following scores on various metrics: BLEU 51.61, METEOR 75.80, TER 37.09, CHRF++ 73.29, BERTScore 92.61, and COMET 76.36. These results demonstrate that fine-tuning multilingual models for a domain-specific machine translation task enhances performance. Our code is released to the public for further exploration https://github.com/Rupeshdhakad06/JUST-NLP-LMT.

1 Introduction

Legal machine translation is more difficult than general translation. It needs both accurate language modelling and correct handling of legal terms (Panezi and O'Shea, 2023). The JUST-NLP 2025 shared task[1] deals with English–Hindi legal translation. The two languages differ in structure and in the way legal contexts are embedded. Accurate translation is not just about replacing words. It also requires keeping the legal meaning and intent the same across both systems (Way, 2016).

Many problems are explored in this area. Using the same legal terms consistently in all contexts is challenging (Altakhaineh, 2025). The lack of parallel legal data limits the amount of supervised training that can be done (Raja and Vats, 2025). In the legal domain, even a small translation mistake can cause serious problems (Llop, 2025). Recent developments in large-scale multilingual NMT, particularly the No Language Left Behind effort, have yielded strong cross-lingual transfer across nearly 200 languages (Costa-jussà et al., 2022, 2024). However, the applicability of these models to specialized domains, such as legal text—especially for Indian languages—remains relatively underexplored (Nair et al., 2024). Within the Indic NLP community, systems such as IndicTrans (Ramesh et al., 2021) and IndicTrans2 (Gala et al., 2023) have broadened multilingual coverage from 11 to 22 languages. Still, hurdles such as rich morphology, multiple scripts, and code-switching persist and complicate model performance on real-world legal corpora (Suman et al., 2023; Sheshadri and Soman, 2023).

This manuscript describes Team-SVNIT's proposed Legal-MT system and experimental evaluation. Our main contributions are: **(1)** An empirical comparison of five candidate translation systems, which identifies the `facebook/nllb-200-distilled-1.3B` model as the best-performing backbone; **(2)** A preprocessing pipeline designed to clean noisy legal text extracted from the Dataset. **(3)** A training regimen using extended contexts (512 tokens), a cautious learning rate (2e-5), and cosine-based scheduling; **(4)** A top-ranking submission that placed first on the task leaderboard; **(5)** A manual, qualitative appraisal of 100 samples to uncover model strengths and recurring error modes.

2 Related Work

2.1 Neural MT for Indic Languages

IndicTrans (Ramesh et al., 2021) was one of the first large-scale, multilingual NMT efforts for 11 Indian languages, employing language-aware pre-

[1] https://exploration-lab.github.io/JUST-NLP/

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 148–154
December 24, 2025

processing and transformer-based architectures. IndicTrans2 (Gala et al., 2023) brought coverage to 22 Indic languages and further refined knowledge distillation methods for scalability. The NLLB (Costa-jussà et al., 2022, 2024) project expanded translation into more than 200 languages with extensive coverage of the Indic families. Distilled versions of these models, 600M and 1.3B parameters, retain impressive translation performance with low computational cost (Koishekenov et al., 2023). The architecture is based on a sparsely gated mixture-of-experts architecture that allows for optimal use of parameters without the computational overhead from dense models.

Despite these advances, some challenges are persistent: the Indic NMT systems have to grapple with morphosyntactic richness, orthographic variations across scripts, limited parallel data, and prevalence of code-switched content (Raja and Vats, 2025; Naveen et al., 2024).

2.2 Legal Domain NMT

Neural machine translation for legal texts differs significantly and is challenging compared to general-domain texts due to the scarcity of domain-specific corpora, specialized terminology, and stringent accuracy demands. Complex syntactic structures in legal texts have to be rendered faithfully, as does the translation of jurisdiction-specific terminology (Way, 2016; Panezi and O'Shea, 2023). Minor mistranslations might have detrimental consequences for legal interpretation and the conduct of proceedings (Llop, 2025).

Altakhaineh et al. show that machine-translated legal content is often fraught with critical semantic and syntactic errors, requiring heavy human post-editing (Altakhaineh, 2025). This emphasizes the importance of domain adaptation, model fine-tuning, and human verification if the systems are to be reliably deployed in a legal context (Princeton, 2025).

2.3 Multilingual Model Fine-tuning

Fine-tuning is a critical step in adapting multilingual models for domain-specific corpora, such as legal text. Cosine annealing with warm restarts helps preserve multilingual prior knowledge and mitigates catastrophic forgetting during training (Loshchilov and Hutter, 2017). Using large batch sizes through gradient accumulation makes training more stable and helps it converge (Han et al., 2024). Parameter-efficient methods, such as LoRA, save resources (Nair et al., 2024). However, full fine-tuning is still better for legal translation, where correctness is more important than speed or cost.

3 Task Description

3.1 Dataset

The JUST-NLP 2025 Legal MT shared task includes an English–Hindi parallel dataset(Singh et al., 2025). It covers different areas of law, such as constitutional, civil, criminal, and administrative. Table 1 shows the statistical details of the dataset. For clarity and to illustrate the nature of the translations, Table 3 in Appendix 4 displays representative English–Hindi sentence pairs selected from the training data.

Split	Pairs	Avg(Eng)	Avg(Hin) in words
Train	50,000	29.3	31.1
Valid	5,000	26.8	30.2
Test	5,000	26.1	–

Table 1: Dataset statistics (average words per sentence).test data translation was not given so mentioned with (–).

The legal sentences in this dataset are long and complex. The average English sentence length in the training data is 29.3 words. The Hindi translations are about 6% longer, averaging 31.1 words. This shows the need for longer context windows during training. The dataset also includes common legal phrases, legal citations, numbers, and some noise from digitization, such as mixed scripts and encoding errors.

4 Dataset Examples

Legal sentences can be challenging to translate. For example, a sentence like *"The appellant, being aggrieved by the judgment and decree dated 15th March 2023 passed by the Hon'ble High Court of Delhi in Civil Appeal No. 2345 of 2022, prefers this present appeal under Section 96 of the Code of Civil Procedure, 1908"* contains numerous legal references, dates, and laws that require careful translation. Latin terms such as *"res judicata"*, *"sub judice"* and *"amicus curiae"* also need proper transliteration and meaning adjustment in Hindi legal language.

Model	Params	Langs	Arch	Features
Helsinki-NLP/opus-mt-en-hi	77M	2	Transformer	Lightweight
facebook/nllb-200-distilled-600M	600M	200	Trans+MoE	Conditional routing
ai4bharat/indictrans2-en-indic-1B	1.0B	22	Transformer	Indic-specialized
law-ai/InLegalTrans-En2Indic-1B	1.0B	Indic	Transformer	Legal domain
facebook/nllb-200-1.3B	1.3B	200	Trans+MoE	Standard version
facebook/nllb-200-distilled-1.3B	**1.3B**	**200**	**Trans+MoE**	**Our choice (distilled)**

Table 2: Model selection prioritizes multilingual capacity, sufficient parameters, and architectural setting.

English	Hindi
according to the learned counsel for the appellant, Allahabad Bank was the owner of the vehicle, as the vehicle in question was unexplained by the Bank.	अपीलार्थी के विद्वान अधि-वक्ता के अनुसार इलाहा-बाद बैंक वाहन का मालिक था क्योंकि प्रश्नाधीन वाहन बैंक के यहां आडमानित था।
both these writ petitions are, thus, allowed.	इस प्रकार, ये दोनों रिट या-चिकाएं अनुज्ञात की जाती हैं।
so far as the applicability of Section 20 is concerned, it is a case of trial.	जहां तक धारा 20 के लागू होने का सवाल है, यह वि-चारण का एक मामला है।

Table 3: Sample training pairs from the legal English–Hindi parallel corpus.

4.1 Evaluation Metrics

The evaluation utilizes six standard translation metrics to assess the model's performance. These are combined into one score called AutoRank. It is defined as:

$$\text{AutoRank} = \frac{1}{6}\sum_{i=1}^{6} M_{i,\text{norm}} \tag{1}$$

The metrics include BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), TER (inverted) (Snover et al., 2006), CHRF++ (Popović, 2017), BERTScore (Zhang et al., 2019), and COMET (Rei et al., 2020). Each score is scaled between 0 and 100.

5 System Architecture

5.1 Model Selection

We tested five translation models, including standard and distilled ones. Table 2 shows the full comparison.

5.2 Preprocessing

There are usually minor errors in Dataset related to legal texts. Our preprocessing pipeline corrected such errors. It substitutes line breaks with spaces and standardized dashes and quotation marks. It also solves the encoding issues, eliminates English words left behind in Hindi text and minimizes the additional spaces. The measures ensure the text remains neat without distorting legal terms, numbers, and references.

Hyperparameter	Value
Base Model	facebook/nllb-200-distilled-1.3B
Max Seq Len	512 tokens
Epochs	20 (early stop)
Batch Size	32
Gradient Accum	16 steps
Effective Batch	512
Learning Rate	2e-5
Scheduler	Cosine w/ restarts
Warmup Ratio	0.1
Optimizer	AdamW
Precision	FP16

Table 4: Training hyperparameters.

5.3 Training Configuration

Table 4 presents our training setup. A context length of 512 covers 99% of the corpus, ensuring full sentence coverage. A large batch size of 512 enables stable optimization, while a conservative learning rate of 2e-5 helps preserve multilingual representations. Cosine scheduling improves convergence by avoiding local minima, and early stopping prevents overfitting. Using FP16 reduces memory usage by 40%, allowing for larger batches and doubling training speed.

5.4 Inference

During inference, the model employs beam search with a width of 4 to identify the optimal translation. The maximum output length is set to 512 tokens, with an n-gram penalty of 3 to avoid repetition. Early stopping ensures efficient decoding once an end token is reached. The process is deterministic, ensuring consistent results, and runs in batches of 64 for faster translation generation.

Model	Params	BLEU	ROUGE	CHRF
Helsinki-OPUS (Without Training)	77M	24.0	50.2	51.3
NLLB-600M + Fine-tuning	600M	43.2	65.5	61.3
ai4bharat/indictrans2-en-indic-1B + Fine-tuning	1.0B	44.0	68.6	62.8
Helsinki-OPUS + Fine-tuning	77M	46.3	70.1	68.9
law-ai/InLegalTrans + Fine-tuning	1.0B	48.1	68.2	66.5
Facebook/NLLB-1.3B + Fine-tuning	1.3B	50.1	73.1	69.4
Facebook/NLLB-1.3B-distilled + Fine-tuning	**1.3B**	**52.1**	**75.6**	**70.9**

Table 5: Validation results sorted by increasing BLEU score. The best-performing setting (distilled NLLB-1.3B + fine-tuning) is shown last for emphasis.

5.5 Computational Requirements

NVIDIA T4 GPU (16GB) on Kaggle: Training time of about 5 hours (prematurely cut off at epoch 12). Memory 14.2GB with FP16. An inferred rate of about 167 sentences per minute. The test set of 5,000 sentences took approximately 6 minutes to complete. Model 2.6GB. Single T4 deployment was made possible with FP16. for faster training, we also used NVIDIA A100.

6 Experiments and Results

6.1 Model Comparison

Table 5 presents the validation results, showing that larger models achieve better performance. The 1.3B model achieves a BLEU score of +7 over the 600M, primarily due to its ability to handle complex legal patterns. Fine-tuning contributes approximately 4.0 BLEU points, which demonstrates its essential role in legal data adaptation. The distilled models are also performing. The `facebook/NLLB-1.3B-distilled` model achieves a score of 52.1 BLEU, compared to the standard version's score of 51. This is because distillation enhances generalization and alleviates overfitting. The general multilingual models are also more effective than the domain-specific models. The NLLB-1.3B-distilled model (52.1 BLEU) outperforms InLegalTrans (48.1 BLEU) due to its more extensive training and multilingual nature.

6.2 Ablation Studies

Table 6 quantifies design choices through ablation studies. The largest factor facilitating domain adaptation is fine-tuning (+4.0). The model size (+6.3) is worth the cost of computation to ensure legality. Complex legal sentences require long context (+1.9). Optimal decoding with beam search (+1.3). N-gram penalty +(0.8) does not allow repetition in legal formulae. Conservative LR (+1.4) maintains the knowledge of multilingualism. Distillation (+0.7) helps to improve performance by refining representations.

Configuration	BLEU	Δ
Full System	**52.1**	—
w/o Fine-tuning	39.2	-4.0
w/ NLLB-600M	43.2	-6.3
Max Length = 256	50.1	-1.9
Beam Width = 1	50.1	-1.3
No n-gram penalty	52.7	-0.8
LR = 2e-5	52.1	-1.4
Standard NLLB-1.3B	52.1	-0.7

Table 6: Ablation results. The distilled variant provides +0.7 BLEU improvement over the standard version.

7 Conclusion & Future Works

We presented Team-SVNIT's winning system for JUST-NLP 2025 Legal MT, achieving 1st place (AutoRank 61.62). Our approach demonstrates that carefully fine-tuned distilled multilingual models (`facebook/NLLB-1.3B-distilled`) outperform both smaller models and domain-specific systems with adequate training data (50K pairs) and systematic optimization. The system incorporates enhanced preprocessing to handle noisy legal texts and an optimized training setup with extended context (512), a conservative learning rate (2e-5), large batches (512), and cosine scheduling for stable convergence. Extensive ablation experiments quantify the impact of these design choices. Overall, the results demonstrate that domain adaptation through fine-tuning remains essential, and that large, well-pretrained multilingual models like NLLB-1.3B-distilled can outperform domain-specific models when sufficient fine-tuning data enables effective adaptation. Future research should address rare terminology through lexical constraints, code-switching through explicit guidelines, very long sentences through hierarchical approaches, multi-reference evaluation for accurate assessment, and document-level translation for improved consistency.

References

Kshetrimayum Boynao Singh, Deepak Kumar, and Asif Ekbal. 2025. Evaluation of LLM for English to Hindi Legal Domain Machine Translation Systems. In *Proceedings of the Tenth Conference on Machine Translation*, pages 823–833, Suzhou, China. Association for Computational Linguistics.

Marta R Costa-jussà, James Cross, Onur Çelebi, et al. 2022. No language left behind: Scaling human-centered translation. *arXiv:2207.04672*.

Marta R Costa-jussà, James Cross, et al. 2024. Scaling neural machine translation to 200 languages. *Nature*, 626:11–18.

Yeskendir Koishekenov, Alexandre Berard, and Vassilina Nikoulina. 2023. Memory-efficient NLLB-200 distilled models. In *Proc. ACL 2023*, pages 8031–8050.

Gowtham Ramesh, Sumanth Doddapaneni, Anoop Kunchukuttan, et al. 2021. Samanantar: Largest parallel corpora for Indic languages. *arXiv:2104.05596*.

Jay Gala, Pranjal A Chitale, Raghavan AK, et al. 2023. IndicTrans2: High-quality MT for 22 Indian languages. *arXiv:2305.16307*.

Argyri Panezi and John O'Shea. 2023. How can we manage the risks and liabilities associated with legal translation in the age of machine translation and generative AI? In *Generative AI and Law Workshop (ICML 2023)*.

Catherine Way. 2016. Challenges of legal translation in 21st century. *Intl. Journal of Communication*, 10:1009–1035.

Abdel Rahman Altakhaineh. 2025. Human versus AI translation of legal documents. *Intl. J. Language and Law*, 14(1):23–47.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: Automatic evaluation method for translation. In *Proc. ACL 2002*, pages 311–318.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: Automatic metric for evaluation. In *Proc. ACL Workshop*, pages 65–72.

Tianyi Zhang, Varsha Kishore, Ruoficer Luan, et al. 2019. BERTScore: Evaluating text with BERT. *arXiv:1904.09675*.

Ricardo Rei, Craig Stewart, Alon Lavie, et al. 2020. COMET: Neural framework for evaluation. In *Proc. EMNLP 2020*, pages 2685–2702.

Maja Popović. 2017. chrF++: Character n-gram F-score. In *Proc. WMT 2017*, pages 612–618.

Matthew Snover, Bonnie Dorr, Richard Schwartz, et al. 2006. Study of translation edit rate evaluation. In *Proc. AMTA 2006*, pages 223–231.

Ilya Loshchilov and Frank Hutter. 2017. SGDR: Gradient descent with restarts. In *Proc. ICLR 2017*.

Linjing Han, Qin Jin, and Michelle Yuan. 2024. Neural MT of clinical text documents. *BMC Medical Informatics and Decision Making*, 24:58.

Rahul Raja and Arpita Vats. 2025. Parallel corpora for low-resource languages. *arXiv:2503.04797*.

P Naveen, Rahul Aralikatte, Anoop Kunchukuttan, et al. 2024. Challenges of MT for low-resource languages. *iScience*, 27(9):110685.

Ashwini R Nair, Sukanta Roy, Swapnil Parekh, et al. 2024. Investigating translation for Indic languages. *Scientific Reports*, 14:17843.

Natalie Llop. 2025. AI, machine translation, and access to justice. *Stanford Law School Working Paper*.

Diptesh Suman, Vivek Srivastava, Swapnil Parekh, et al. 2023. Machine translation for low-resource languages. In *Proc. WMT 2023*, pages 856–864.

Shailashree K Sheshadri and KP Soman. 2023. A voyage on neural machine translation for Indic languages. *Procedia Computer Science*, 218:2694–2712.

Soham Bhattacharjee, Baban Gain, and Asif Ekbal. 2024. Domain dynamics: LLMs in English-Hindi translation. In *Proc. WMT 2024*, pages 341–354.

Markus Freitag, Nitika Mathur, Daniel Deutsch, et al. 2024. Are LLMs breaking MT metrics? WMT24 analysis. In *Proc. WMT 2024*, pages 47–81.

Ashish Vaswani, Noam Shazeer, Parmar Nikita, et al. 2017. Attention is all you need. In *Proc. NeurIPS 2017*, pages 5998–6008.

Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling knowledge in neural networks. *arXiv:1503.02531*.

Baban Gain, Ramakrishna Appicharla, Soumya Chennabasavaraj, et al. 2022. Low resource chat translation: Hindi-English benchmark. In *Proc. AMTA 2022*, pages 83–96.

Ramakrishna Appicharla, Asif Ekbal, and Pushpak Bhattacharyya. 2021. EduMT: MT system for educational content. In *Proc. ICON 2021*, pages 35–43.

Princeton NLP Group. 2025. Managing Context in Low-Resource Legal Machine Translation: A Case Study in English-Hindi. In *Proceedings of JUST-NLP 2025 Shared Task*.

Pratik Joshi, Sumanth Paul, Mukund Sharma, et al. 2024. IL-TUR: Benchmark for Indian legal understanding. *arXiv:2407.05399*.

Sayan Mahapatra, Debtanu Datta, Shubham Soni, et al. 2024. MILPaC: Benchmark for legal text translation. *ACM TALLIP*.

Ramakrishna Appicharla, Baban Gain, Santanu Pal, and Asif Ekbal. 2025. Beyond the sentence: Context-aware MT survey. *arXiv:2506.07583*.

Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, et al. 2024. Preliminary WMT24 ranking of MT systems. *arXiv:2407.19884*.

Appendix

A Result Analysis

A.1 Qualitative Analysis through Translation Examples

To better understand the performance of our model, we conducted a detailed manual analysis of the translation outputs. Table 7 presents representative examples showcasing both strengths and limitations.

The examples reveal several important patterns. Our model excels at translating standard legal terminology and complex sentence structures, particularly for common legal procedures and statutory references. The preservation of numerical data is consistently better. However, challenges remain with Latin legal terms (*quantum meruit*) and specialized legal concepts that require cultural adaptation rather than direct transliteration.

Nearly half of the translation mistakes occurred because the model couldn't handle unusual legal terms, particularly Latin phrases and specific legal concepts. This often means the system didn't "understand" a word, translated it incorrectly, or omitted it if it wasn't common in its training data. These problems arise frequently in legal writing, where precise terminology is crucial. To determine this percentage, we reviewed all mistakes in a batch of sample translations and counted the number that involved rare terminology. To reduce these errors, you'll need better resources or databases for legal terms, so the model knows what they mean. The next biggest issue (20%) came from the system mixing up pronouns in sentences with multiple people or actors, which shows it sometimes "loses track" of who is being referred to in complicated legal sentences.

A.2 Comparative Advantage of Distilled Models

Many reasons justify the superior performance of the distilled NLLB-1.3B variant, +0.7 BLEU over the standard variant(facebook/nllb-200-1.3B). First, knowledge distillation during pretraining definitely forces the model to learn more generalized representations rather than memorizing training patterns. Second, distilled models exhibit better calibration and reduced overconfidence, which is a crucial requirement for legal translation in accurately representing uncertainty. Third, the distillation process appears to enhance cross-lingual transfer efficiency, which is particularly beneficial in the case of English-Hindi legal translation due to the limited amount of available parallel data.

A.3 Practical Implications and Deployment Considerations

Our findings have significant practical consequences for legal translation workflows. The low TER score of 37.09 indicates that post-editing effort would be considerably reduced, allowing translator productivity to increase 2-3×. Numerical data and citations are perfectly preserved, which eliminates critical risks in legal documentation. The 40 to 45 % error rate confirms that human review is still important for legally binding documents.

The model size is moderate at 2.6GB, with computational demands for deployment in-house, which addresses all data confidentiality concerns usually associated with legal practice. The relatively short training time of under 3 hours enables organizations to fine-tune the model for their specific legal subdomains, such as patent law or corporate contracts.

English Source	Hindi Translation
the appellant is acquitted.	अपीलार्थी को बरी किया जाता है
they also raised memorials on the merits and the preliminary habit.	उन्होंने गुणावगुणों तथा प्रारंभिक अभ्यापत्ति पर भी सम्प–रीक्षण किये थे ।
being aggrieved by the order dated 2nd March , 2012 made by the learned Single Judge in CWJC No.3653 of 2012 , the writ petitioner has filed this appeal under clause 10 of the Letters Patent .	CWJC सं ० 3653 वर्ष 2012 में विद्वान एकल न्याया–धीश द्वारा किये गये दिनांक 2 मार्च, 2012 के आदेश से व्यथित होकर, रिट याची ने लेटर्स पेटेंट के खंड 10 के अधीन यह अपील दाखिल किया है ।
6- The opposition no.2 has filed his presence in this Court by filing the right in favour of his learned counsel , though he has not filed any counter affidavit .	6 – विपक्षी संख्या 2 ने अपने विद्वान अधिवक्ता के पक्ष में अधिकार दाखिल करके इस न्यायालय में अपनी उपस्थिति दर्ज करायी है, यद्यपि उसने कोई प्रति शपथ पत्र दाखिल नहीं किया है ।
7- We have heard the counsel for the learned Principal Additional Advocate General, Muzaffarpur Properties Private Limited, Smt. Shahida Hassan and the counsels of various dignitaries who have filed applications in these appeals both on facts as well as on law.	7 – हमने विद्वान प्रधान महाधिवक्ता , मुजफ्फरपुर गुण प्राईवेट लिमिटेड , श्रीमती शाधा हसन के अधिवक्ताओं को सुना है जिन्होंने दोनों तथ्यों तथा तथ्यों पर भी तथा विधि पर भी इन अपीलों में आवेदन दाखिल किये हैं ।
the aforesaid case was of the Central Excise Act and section 35H of the Central Excise Act provided that an appeal and reference should be made to the High Court within 180 days from the date of communication of the judgment of the order.	पूर्वोक्त मामला केन्द्रीय उत्पाद अधिनियम की केंद्रीय उत्पाद अधिनियम एवं धारा 35H का था यह प्रावधान करता है कि आदेश के निर्णय की संसूचना की तिथि से 180 दिनों के भीतर उच्च न्यायालय को एक अपील एवं निर्देश दिया जाना चाहिए ।

Table 7: English-Hindi translation examples demonstrating model performance.

Combining Extractive and Generative Methods for Legal Summarization: Tayronas Trigrams at JUST-NLP 2025

Erich Giusseppe Soto Parada
Universidad de los Andes
Bogotá, Colombia
eg.soto@uniandes.edu.co

David Cuevas Alba
Universidad de los Andes
Bogotá, Colombia
d.cuevas@uniandes.edu.co

Carlos Manuel Muñoz Almeida
Universidad de los Andes
Bogotá, Colombia
c.munoza@uniandes.edu.co

Abstract

This paper presents Tayronas Trigrams's methodology and findings from our participation in the JUST-NLP 2025 Shared Task of Legal Summarization (L-SUMM), which focused on generating abstractive summaries of lengthy Indian court judgments. Our initial approach involved evaluating and fine-tuning specialized sequence-to-sequence models like Legal-Pegasus, Indian Legal LED, and BART. We found that these small generative models, even after fine-tuning on the limited InLSum dataset (1,200 training examples), delivered performance (e.g., Legal-Pegasus AVG score: 16.50) significantly below expected.

Consequently, our final, best-performing method was a hybrid extractive-abstractive pipeline. This approach first employed the extractive method PACSUM to select the most important sentences yielding an initial AVG score of 20.04 and then utilized a Large Language Model (specifically Gemini 2.5 Pro), correctly prompted, to perform the final abstractive step by seamlessly stitching and ensuring coherence between these extracted chunks. This hybrid strategy achieved an average ROUGE-2 of 21.05, ROUGE-L of 24.35, and BLEU of 15.12, securing 7th place in the competition. Our key finding is that, under data scarcity, a two-stage hybrid approach dramatically outperforms end-to-end abstractive fine-tuning on smaller models.

1 Introduction

The legal systems of highly populous nations, such as India, are facing a critical challenge due to judicial pendency. As reported by the National Judicial Data Grid (NJDG), India alone contends with over 44 million pending cases across its courts. This massive backlog, often caused by manual, inefficient document processing, delays timely justice and undermines the fundamental rights the system is designed to protect. Automated systems powered by Natural Language Processing (NLP) offer a scalable solution to assist legal professionals, streamline document workflow, and ultimately improve public access to case information.

Our team, Tayronas Trigrams, participated in the JUST-NLP 2025 Shared Task 1: Legal Summarization (L-SUMM), which required generating 500-word abstractive summaries for lengthy Indian court judgments. We found that the task's inherent challenges, such as nuanced legal reasoning and textual abstraction, were significantly amplified by the limited InLSum dataset (1,200 training instances). This data scarcity caused existing specialized small generative models to struggle, motivating our pivot from a purely abstractive approach to a novel, two-stage methodology.

- We demonstrate the limited efficacy of fine-tuning small, specialized transformer models (e.g., Legal-Pegasus, Indian Legal LED, BART) for abstractive legal summarization under low-resource conditions.
- We establish that a simple extractive baseline significantly outperforms these fine-tuned generative models when data is scarce.
- We propose and validate a Hybrid Extractive-Abstractive pipeline, which uses Large Language Models (LLMs) to connect and coherently refine extracted content, resulting in a substantial performance gain.
- Our final model achieved strong competitive metrics (ROUGE-2: 21.05, ROUGE-L: 24.35) by leveraging this hybrid approach, underscoring its superiority for low-data, domain-specific summarization.

2 Related Work

Automatic legal summarization is challenging due to extreme document length (Shukla et al., 2022),

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 155–161
December 24, 2025

domain-specific "legalese" (Joshi et al., 2024), and the need for high factual consistency. Indian legal texts present additional challenges, being "noisier and poorly organized" (Sharma et al., 2023). While traditional extractive methods like LexRank are factually robust (Shukla et al., 2022; Sharma et al., 2023), recent work has shifted to abstractive models like BART (Shukla et al., 2022) and specialized models such as Legal-Pegasus. Notably, Sharma et al. (2023) identified Legal-Pegasus as a top performer on Indian legal data, justifying its use as a strong baseline.

A primary obstacle, however, is data scarcity. The L-SUMM task utilizes the low-resource InL-Sum dataset, creating a significant performance bottleneck. This setting favors robust baselines; Shukla et al. (2022) observed that strong extractive models can perform on par with abstractive ones. Furthermore, Joshi et al. (2024) found that even large LLMs (e.g., GPT-4) can underperform fine-tuned models on this specific task (SUMM). Our work builds directly on these findings, corroborating the limitations of fine-tuning small models in a low-resource setting and instead proposing a hybrid approach that combines extractive content selection with LLM-based refinement.

3 Methodology

Our methodology was structured as a multi-stage process, beginning with a systematic evaluation of existing models and culminating in a hybrid approach. The initial phase, detailed below, focused on benchmarking specialized pre-trained models to establish a performance baseline on the L-SUMM task, following similar comparative analyses in the literature (Sharma et al., 2023; Shukla et al., 2022). This foundational analysis was critical in guiding our subsequent, more complex strategies.

3.1 Initial Model Evaluation

To assess the zero-shot capabilities of existing models on the provided dataset, we implemented a unified evaluation pipeline using the Hugging Face Transformers library. This allowed us to systematically compare promising pre-trained, domain-specific models for legal text summarization (Sharma et al., 2023; Shukla et al., 2022). The selected models were:

- **BART** (`sanatann/legal-summarizer-bart`): A model based on the BART architecture, fine-tuned for legal summarization, commonly used as a baseline for this task (Sharma et al., 2023; Shukla et al., 2022).
- **LED** (`TheGod-2003/legal-summarizer`): A Longformer-Encoder-Decoder (LED) model designed to handle long documents, making it suitable for legal texts (Sharma et al., 2023; Shukla et al., 2022).
- **Indian Legal LED** (`Yashaswat/indian-legal-led-base`): An LED model specifically fine-tuned on Indian legal documents. The use of a specialized Legal-LED is supported by Joshi et al. (2024) and Shukla et al. (2022), who identify it as a strong-performing model for the domain.
- **Legal Pegasus** (`nsi319/legal-pegasus`): A model based on the Pegasus architecture adapted for the legal domain. This model was prioritized for evaluation as Sharma et al. (2023) found it to “outperform over all other models” for Indian legal summarization, a finding supported by Shukla et al. (2022).

Our pipeline loaded each model and its corresponding tokenizer, processed documents from the training set, generated summaries, and evaluated them against the ground truth. Performance was measured using standard summarization metrics: ROUGE-2, ROUGE-L, and BLEU, which are standard for the task and the In-Abs dataset (Joshi et al., 2024; Shukla et al., 2022). This initial evaluation provided the baseline data that informed our decision to move away from a purely fine-tuning-based approach.

3.1.1 Initial Evaluation Results

The initial evaluation was conducted on a small, representative sample of 8 documents from the training set to quickly gauge the zero-shot performance of each model. While not statistically significant, this preliminary analysis provided valuable directional insights into which models were most promising. The results, shown in Table 1, indicated that Legal Pegasus offered a competitive starting point.

3.2 Fine-Tuning Phase

3.2.1 Motivation and Model Selection

Following Bhattacharya et al. (2021), who recommended Legal-Pegasus for legal summarization via

Model	ROUGE-2	ROUGE-L	BLEU
BART	9.34	17.37	10.06
LED	11.77	18.91	16.30
Indian Legal LED	2.45	16.67	7.19
Legal Pegasus	20.05	17.37	10.06

Table 1: Preliminary zero-shot results on a sample of 8 documents. Scores are corpus-level.

chunking, we selected `nsi319/legal-pegasus` as our base model. However, Indian legal judgments averaged 10,000+ tokens, exceeding Legal-Pegasus's 1,024-token limit, necessitating an intelligent chunking strategy. We investigated whether embedding-based similarity could improve chunk-summary alignments.

3.2.2 Chunking Strategy

We adopted the chunking approach of Bhattacharya et al. (2021), inspired by Gidiotis and Tsoumakas (2020), which segments documents into chunks and constructs targeted summaries by mapping summary sentences to similar document sentences. Given a document-summary pair (d, s), we partition d into chunks $\{d_1, d_2, \ldots, d_n\}$ and generate chunk-specific summaries s_i by aggregating summary sentences that map to sentences within each chunk d_i (see Appendix A for detailed methodology). We compared two similarity metrics while keeping the fine-tuning pipeline constant.

Experiment 1: TF-IDF Baseline. Our baseline employed TF-IDF cosine similarity with fixed-size chunks (400 words, 50 overlap) and a 0.1 similarity threshold, yielding 4,706 training pairs. While computationally efficient, this bag-of-words approach lacks semantic context (detailed vectorization parameters in Appendix A).

Experiment 2: MCS-SBERT Optimization. Building on Mean Cosine Similarity (MCS) (Reimers and Gurevych, 2019), we replaced TF-IDF with dense embeddings from `all-MiniLM-L6-v2` Sentence-BERT (384-dim), which capture semantic relationships beyond lexical overlap. We implemented semantic chunking respecting paragraph boundaries (500 words, 40-word minimum) and raised the similarity threshold to 0.4 to filter false positives. Quality filters (compression ratio 0.05–0.4, minimum summary length 40 words) excluded degenerate pairs, yielding 638 high-confidence examples—a deliberate trade-off sacrificing quantity for quality.

Analysis. Table 2 shows MCS-SBERT outperformed TF-IDF with 82% fewer pairs, attributable to semantic capture (SBERT encodes meaning beyond surface similarity) and noise reduction (elevated threshold filtered spurious alignments). This validates that in low-resource legal domains, curated high-precision data outweighs large volumes of noisy examples.

3.2.3 Fine-tuning Setup

We fine-tuned Legal-Pegasus using Hugging Face Transformers with the 638 MCS-SBERT pairs (85/15 split). Configuration: LR 2×10^{-5}, 500 warmup steps, batch 4 (effective: 8), 3 epochs, 1,024/512 tokens, BF16 on A100 GPU, beam search (6). Training converged in 45 minutes with ROUGE evaluation (Lin, 2004).

3.2.4 Results and Analysis

Our fine-tuned Legal-Pegasus achieved ROUGE-2: 17.1 and ROUGE-L: 19.8, validating MCS-SBERT's effectiveness but falling below our extractive baseline (ROUGE-2: 20.51, ROUGE-L: 23.49, as shown in Table 3). This gap reveals limitations of fine-tuning small models in low-resource legal scenarios: (1) *data scarcity*—638 pairs vs. 10,000+ typically required (Zhang et al., 2020); (2) *domain complexity*—legal precision demands exceed small model capacity (568M parameters), causing occasional hallucinations; and (3) *abstractive risk*—semantic drifts unacceptable for legal fidelity, where extractive methods excel.

Despite suboptimal performance, this validated dense embeddings' superiority over bag-of-words and demonstrated fine-tuning's viability threshold, motivating our pivot to a hybrid extractive-abstractive approach.

3.3 Hybrid Extractive-Abstractive Method

We propose a two-stage approach combining extractive pre-selection with abstractive refinement for legal document summarization. Legal case judgments pose significant challenges due to extreme length (10,000+ tokens), technical terminology, and complex argumentation structures (Bhattacharya et al., 2021).

3.3.1 Final method

Stage 1: Extractive Pre-selection. Documents are first segmented into semantic chunks (maximum 512 tokens per chunk, no overlap), breaking at natural boundaries to preserve meaning. We apply PACSUM (Zheng and Lapata, 2019), a BERT-

Table 2: Comparative study of chunking strategies. MCS-SBERT achieves superior performance with 82% fewer training pairs. The performance is reported on the InLSum Validation Set (200 datapoints).

Method	Similarity	Threshold	Pairs	AVG	R-2	R-L	BLEU
TF-IDF Baseline	Cosine (BoW)	0.1	4,706	16.77	19.94	22.51	7.85
MCS-SBERT	SBERT Cosine	0.4	638	**17.16**	**20.64**	**21.62**	**9.23**

based unsupervised method that constructs directed sentence graphs with position-augmented centrality scoring. PACSUM ranks sentences within each chunk by semantic similarity and positional importance. We then select the chunks corresponding to the highest PACSUM scores and aggregate them sequentially until reaching a 1000-token budget, adding additional highly-ranked content when space permits. This reduces input length by 60-70% while retaining salient information.

Stage 2: Abstractive Refinement. The aggregated 1000-token extractive summary is processed by Gemini 2.5 Pro (DeepMind, 2024) using zero-shot prompting with optimized instructions. We employ automated prompt optimization through iterative refinement, evaluating candidate prompts on validation examples and selecting those maximizing ROUGE scores. While we experimented with few-shot learning (1-3 demonstration examples), zero-shot prompting consistently outperformed few-shot across all metrics. The LLM generates coherent abstractive summaries by paraphrasing, fusing sentences, and removing redundancy.

3.3.2 Experimental Results

We evaluate our method on the Indian Court Judgment Summarization shared task. Table 3 presents performance across four metrics: the AVG Score (primary ranking metric), ROUGE-2 ($R2$), ROUGE-L (RL), and BLEU (B). The AVG Score (C) is calculated as the average of the three standard relevance metrics:

$$\text{AVG Score } (C) = \frac{R2 + RL + B}{3}$$

Method	Score	R-2	R-L	BLEU
PACSUM (extractive)	19.61	20.51	23.49	14.84
PACSUM + Gemini 2.5 Pro	**20.17**	**21.05**	**24.35**	**15.12**

Table 3: Performance comparison on test set. PACSUM processes 512-token chunks and aggregates to 1000 tokens. The hybrid method shows consistent improvements across all metrics.

The hybrid approach outperforms the purely extractive baseline across all metrics, with improvements of +0.56 points in Competition Score, +0.54 in ROUGE-2, +0.86 in ROUGE-L, and +0.28 in BLEU. These gains demonstrate that abstractive refinement by the LLM adds value beyond extractive selection alone.

Analysis. The modest improvements suggest that the extractive pre-selection already captures most salient content effectively. The LLM's primary contribution is enhancing fluency and coherence rather than identifying additional key information. The two-stage strategy (512-token chunks for processing, 1000-token aggregation for LLM input) proved critical for managing computational costs while maintaining summary quality. Notably, zero-shot prompting outperformed few-shot approaches, suggesting that well-crafted instructions are more effective than demonstration examples for this task.

4 Conclusion

In this paper, we described the methodology used by Tayronas Trigrams for the L-SUMM task at JUST-NLP 2025. Our initial experiments focused on fine-tuning specialized models like Legal-Pegasus, but we found that they did not perform well due to the small size of the InLSum dataset (1,200 examples). The models struggled to generate accurate summaries without more training data.

To solve this, we developed a hybrid pipeline. We used PACSUM for extractive selection to identify the most relevant sentences, followed by Gemini 2.5 Pro for abstractive refinement. Our approach achieved 7th place in the competition with a ROUGE-L score of 24.35. These results indicate that when data is limited, combining extractive methods with Large Language Models is a practical and effective strategy for legal summarization. Future work will explore dynamic chunking strategies to mitigate information bottlenecks and evaluate open-source models to address the privacy limitations of proprietary APIs.

Limitations

While our hybrid extractive-abstractive approach proved effective for the shared task, we acknowledge three primary limitations:

- **Dependency on Proprietary APIs:** Our reliance on Gemini 2.5 Pro introduces costs and potential reproducibility issues. Furthermore, sending legal documents to external APIs raises data privacy concerns that purely local models avoid.
- **Information Bottleneck:** The abstractive generator is strictly limited by the initial extractive step. If the PACSUM algorithm fails to select a crucial piece of evidence or legal precedent, the LLM has no way to recover that information, as it never sees the full original document.
- **Domain Specificity:** Our optimization steps, particularly the chunking thresholds and prompt engineering, were tailored specifically for Indian case law. These parameters may not generalize effectively to other legal systems or languages without significant adjustment.

References

Paheli Bhattacharya, Kaustubh Hiware, Subham Rajgaria, Nilay Pochhi, Kripabandhu Ghosh, and Saptarshi Ghosh. 2021. A comparative study of summarization algorithms applied to legal case judgments. In *Proceedings of the 43rd European Conference on Information Retrieval*, pages 413–428. Springer.

Google DeepMind. 2024. Gemini: A family of highly capable multimodal models. `https://blog.google/technology/ai/google-gemini-next-generation-model-february-2024/`. Accessed: 2024-12-15.

Alexios Gidiotis and Grigorios Tsoumakas. 2020. A divide-and-conquer approach to the summarization of long documents. In *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, volume 28, pages 3029–3040. IEEE.

Abhinav Joshi, Shounak Paul, Akshat Sharma, Pawan Goyal, Saptarshi Ghosh, and Ashutosh Modi. 2024. IL-TUR: Benchmark for Indian legal text understanding and reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11460–11499, Bangkok, Thailand. Association for Computational Linguistics.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Saloni Sharma, Surabhi Srivastava, Pradeepika Verma, Anshul Verma, and Sachchida Nand Chaurasia. 2023. A comprehensive analysis of indian legal documents summarization techniques. *SN Comput. Sci.*, 4(5).

Abhay Shukla, Paheli Bhattacharya, Soham Poddar, Rajdeep Mukherjee, Kripabandhu Ghosh, Pawan Goyal, and Saptarshi Ghosh. 2022. Legal case document summarization: Extractive and abstractive methods and their evaluation. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1048–1064, Online only. Association for Computational Linguistics.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119, pages 11328–11339. PMLR.

Hao Zheng and Mirella Lapata. 2019. Sentence centrality revisited for unsupervised summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6236–6247, Florence, Italy. Association for Computational Linguistics.

Appendix

A Chunking Strategy: Detailed Methodology

Our chunking approach follows the methodology of Bhattacharya et al. (2021), inspired by Gidiotis and Tsoumakas (2020), which addresses a fundamental challenge: pre-trained models have input length limits shorter than legal documents, yet naively using the same reference summary for all chunks ignores their varying content.

A.1 Chunk-Specific Summary Generation

Given a training pair (d, s) where d is a document and s is its reference summary:

Step 1: Document Segmentation. Partition d into n chunks: $d = \{d_1, d_2, \ldots, d_n\}$.

Step 2: Sentence-Level Mapping. For each sentence s_j in the reference summary s, identify its most similar sentence in the document d:

$$\text{map}(s_j) = \arg\max_{d_k \in d} \text{sim}(s_j, d_k) \tag{1}$$

where $\text{sim}(\cdot, \cdot)$ is a sentence similarity measure (detailed below).

Step 3: Chunk-Specific Summary Construction. For each chunk d_i, construct its target summary s_i by aggregating all summary sentences mapped to sentences within that chunk:

$$s_i = \{s_j \in s \mid \text{map}(s_j) \in d_i\} \tag{2}$$

This procedure generates multiple training pairs (d_i, s_i) from each document, where each chunk is paired with a semantically relevant subset of the original summary.

A.2 Similarity Measures

We compare two sentence similarity metrics:

A.2.1 TF-IDF Cosine Similarity (Baseline)

Representation. Sentences are represented as sparse TF-IDF vectors. We apply preprocessing (tokenization, stopword removal including legal terms, stemming) and vectorize using:

- 3,000-dimensional vocabulary
- Unigram + bigram features
- Sublinear TF scaling: $\text{tf}(t, d) = 1 + \log(\text{count}(t, d))$

Similarity. For sentences s_j (summary) and d_k (document) with TF-IDF vectors $\mathbf{v}_s$ and $\mathbf{v}_d$:

$$\text{sim}_{\text{TF-IDF}}(s_j, d_k) = \frac{\mathbf{v}_s \cdot \mathbf{v}_d}{\|\mathbf{v}_s\| \|\mathbf{v}_d\|} \tag{3}$$

Chunking Parameters.

- Fixed sliding window: 400 words, 50-word overlap
- Similarity threshold: 0.1 (sentence s_j maps to d_k if similarity ≥ 0.1)

This lexical baseline captures term overlap but ignores semantic context.

A.2.2 Mean Cosine Similarity with SBERT (MCS)

Representation. We use `all-MiniLM-L6-v2` Sentence-BERT (Reimers and Gurevych, 2019) to encode sentences as 384-dimensional dense embeddings. Unlike TF-IDF, SBERT operates on raw text and captures contextual semantics.

Similarity. For embeddings $\mathbf{e}_s, \mathbf{e}_d \in \mathbb{R}^{384}$:

$$\text{sim}_{\text{MCS}}(s_j, d_k) = \frac{\mathbf{e}_s \cdot \mathbf{e}_d}{\|\mathbf{e}_s\|_2 \|\mathbf{e}_d\|_2} \tag{4}$$

Chunking Parameters.

- Semantic chunking: respects paragraph boundaries, 500-word target size
- Similarity threshold: 0.4 (raised to mitigate false positives from dense embeddings)
- Quality filters: compression ratio $\in [0.05, 0.4]$, minimum 40 words

The higher threshold and quality filters prioritize precision over recall, trading dataset size for semantic coherence.

A.2.3 DSPy Prompt Configuration

The optimized prompt used in our hybrid extractive-abstractive pipeline:

> **Instructions:** Create an extractive summary of a legal judgment by identifying and concatenating key paragraphs.
>
> **Important Notes:**
>
> - If example summaries are shown above, they are ONLY for demonstrating format and style
> - Do NOT use factual content, names, or legal arguments from examples
> - ONLY summarize the specific judgment provided in the "Judgment Text" field
> - Each judgment is separate - do not mix information between cases
>
> **Instructions:**
>
> 1. Identify key paragraphs covering: case parties/background, core legal issue, court's decision, and key reasoning
> 2. Optimize for ROUGE-2, ROUGE-L and BLEU by copying text as exactly as possible
> 3. Join paragraphs in logical order with minimal transitions
> 4. Preserve original legal terminology, case citations, and phrasing
>
> **Target:** 500-700 words from current judgment only

Prompt Structure:

- **Input:** Full judgment text preceded by "— NEW JUDGMENT —"
- **Reasoning:** Chain-of-thought with "Let's think step by step in order to"
- **Output:** Extractive summary preceded by "— SUMMARY OUTPUT —"

Automatic Legal Judgment Summarization Using Large Language Models: A Case Study for the JUST-NLP 2025 Shared Task

Santiago Chica
s.chica10@uniandes.edu.co
Universidad de los Andes
Bogotá, Colombia

Abstract

This paper presents the proposal developed for the **JUST-NLP 2025 Shared Task on Legal Summarization**, which aims to generate abstractive summaries of Indian court judgments. We describe the motivation, dataset analysis, related work, and proposed methodology based on Large Language Models (LLMs). We analyze the Indian Legal Summarization (InLSum) dataset, review four relevant articles in the summarization of legal texts, and describe the experimental setup involving GPT-4.1 to evaluate the effectiveness of different prompting strategies. The evaluation will follow the ROUGE and BLEU metrics consistent with the competition protocol.

1 Introduction

The problem addressed is the automatic summarization of legal documents, specifically Indian court judgments. This task is part of the *JUST-NLP 2025 Shared Task on Legal Summarization*, which evaluates models generating abstractive summaries from 1,200 training and 200 validation cases, later applied to 400 unseen test cases. Performance is measured using ROUGE-2, ROUGE-L, and BLEU. The goal of this project is to design and evaluate LLM-based systems that maximize these metrics.

2 Problem Statement and Motivation

Legal professionals in Common-Law systems must review extensive judgments to identify precedents. Manual review is time-consuming (Shukla et al., 2022). Automatic summarization can effectively reduce this considerable effort, but legal texts are long, syntactically complex, and filled with citations, and domain-specific terminology (Sharma et al., 2023). Our proposal introduces a novel approach to this domain within the context of Indian High Court judgments, specifically by exploring diverse prompting strategies and hybrid extractive-abstractive pipelines to enhance the factual accuracy of LLM-generated summaries.

3 Dataset Description and Analysis

We use the **InLSum** dataset provided by the competition, containing:

- **Train:** 1,200 judgments and 1,200 human-written summaries.
- **Validation:** 200 judgments.
- **Test:** 400 judgments (released later).

Each file is in JSONL format:

```
{"ID": "id_100", "Judgment": "<text>"}
{"ID": "id_100", "Summary": "<ref. summary>"}
```

3.1 Descriptive Statistics

	Judgments	Summaries
Avg. words	7,418	545
Median	2,940	516
Min / Max	159 / 134,483	26 / 2,083
Compression ratio	26% (avg), 18% (median)	

Table 1: Descriptive statistics of the InLSum dataset.

Documents show large variance: judgments are heterogeneous and lengthy, while summaries are more uniform and predictable. Lexical patterns (case numbers, articles, sections, petitioner/respondent, dates) appear consistently in both sets, confirming structural alignment.

4 Related Work

4.1 Architectural Advances for Long Legal Summarization

Several transformer architectures have been influential for long-document summarization tasks. Lewis et al. (2020) proposed **BART**, a denoising sequence-to-sequence pre-training approach that remains a key baseline for generative summarization.

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 162–170

Beltagy et al. (2020) introduced the **Longformer** model, designed for long-context encoding through sliding-window attention mechanisms, which significantly improves scalability on multi-thousand-token legal texts. Similarly, Bajaj et al. (2021) explored low-resource long-document summarization using pretrained language models, providing valuable insights into resource-efficient setups relevant for the Indian legal domain.

4.2 Benchmarking and Domain-Specific Adaptation

Prior work has established benchmarks for both general-purpose and domain-adapted models on legal text. Datta et al. (2023) introduced **MILDSum**, a bilingual English–Hindi legal corpus for summarization, enabling cross-lingual evaluation. Their work demonstrates that domain-specific datasets built with legal rigor improve supervised and abstractive training quality in multilingual contexts. Sharma et al. (2023) conducted a comprehensive comparison of BART, Longformer, and Legal-Pegasus on Indian court judgments. Their study found that Legal-Pegasus achieved the highest ROUGE-L score of approximately 0.3, showing that pretrained legal models outperform general-purpose models when fine-tuned. Furthermore, Shukla et al. (2022) benchmarked multiple extractive and abstractive methods, including SummaRuNNer, Legal-Pegasus, and Longformer, on Indian case law, emphasizing the effectiveness of chunking and hybridization techniques for summarizing long and complex legal judgments.

4.3 LLMs, Factuality, and Hybrid Pipelines

The recent application of LLMs has introduced new challenges and opportunities. Deroy et al. (2024) presented one of the first evaluations of GPT-3.5 and GPT-4 against domain-specific legal models. They found that while LLMs outperform traditional extractive baselines, they often hallucinate or omit key details, highlighting the need for hybrid extractive–abstractive pipelines and chunking strategies to maintain factual consistency. The cited work, however, did not explore the role of prompting strategies in improving performance or avoiding hallucination. Our proposal is intended to delineate the potential of this important LLM characteristic.

4.4 Summary of Key Findings and Gaps

Across these studies, several consistent findings emerge:

- Domain adaptation and legal-specific corpora improve the ROUGE and BLEU metrics.
- Hybrid extractive–abstractive designs mitigate hallucination and improve factual faithfulness.
- Attention-efficient transformers and LLMs now define the state of the art for long legal texts.

Crucially, the systematic investigation of **prompting strategies** as a method to control LLM factuality and performance remains an underexplored gap, which this work addresses.

5 Proposed Methodology

5.1 Overview

We follow a quantitative experimental design using LLMs with structured prompting and multi-agent flows.

5.2 Prompting Strategies

Prompt Families We designed and evaluated several prompt families:

- **Simple baseline**: Employing "*Tl;Dr*" as a simple prompt to establish a comparative starting point for evaluating more complex instruction sets. (The baseline prompt can be found on Appendix A.4)
- **Few-shot**: Introduces the task with a focus on metric maximization, followed by an outline that specifies the target compression ratio, sentence length, and legal term preservation ratio. Three example judgment/summary pairs are then supplied to guide the model on the correct structure, phrasing, order, and span length of the final output.
- **Reward System (Winning Prompt)**: This advanced prompt implements a comprehensive, gamified scoring system designed to explicitly reward factually precise and structurally correct outputs. It uses progressive rewards for copying long exact sequences, applies contextual multipliers based on sentence placement, and integrates density targets for key lexical features. Crucially, it enforces structural excellence through specific bonuses and heavily penalizes hallucination and paraphrasing of critical legal terminology. This prompt was generated following a heuristic approach,

starting with strategies focused on improving contest metrics and iterating on changes that further increased those metrics. (The complete scoring system prompt is provided in Appendix A.1).

- **Multi-Agentic approach**: We implemented multiple structured prompts, each set up for a specific subtask, to help extract the text's individual features for analysis (details follow in the next section).

5.3 Multi-Agent Architecture

We implemented multiple architectural approaches, to explore different strategies for legal summarization. Each approach uses a state graph with autonomous agents handling different sub-tasks, and an orchestrator managing the workflow.

Approach 1 – Basic Extract/Abstract Pipeline: A two-stage setup. *Extraction* (see Appendix A.2 for the complete prompt) copies literal spans into a structured schema (court, date, parties, counsel, facts, issues, arguments, decision, reasoning, orders, citations). *Abstraction* (see Appendix A.3 for the complete prompt) assembles the final summary by concatenating verbatim phrases (typically 450–550 words), prioritizing long n-grams and exact terminology to maximize ROUGE-L and BLEU. Our approach is inspired by Deroy et al. (2024), but our methodology employs LLMs for both stages of the summarization pipeline. This contrasts with the cited work, which utilized established extractive techniques, i.e., CaseSummarizer, BertSum, and SummaRunner, in its initial extractive phase.

Approach 2 – Domain-Aware Pipeline: This methodology implements a three-stage pipeline designed to mitigate the inherent loss of factuality and context in the summarization task. The process is developed as follows:

1. **Domain Classification:** Initially, an LLM agent classifies the judgment into a specific legal area (criminal, civil, constitutional, etc.). This classification enables the retrieval of domain-specific structural and statistical patterns (canonical section order, average section lengths), which act as structural guides for the subsequent stages (see Appendix A.5 for the full prompt).

2. **Domain-Aware Structured Extraction:** A structured extraction prompt is applied to segment the original text and copy literal fragments of the judgment (party names, facts, arguments, decisions) into a JSON format. This stage is crucially extractive to ensure lexical fidelity (*i.e.*, maximizing ROUGE-L and BLEU). The LLM is instructed to follow the section order and typical length guidelines of the domain identified previously (see Appendix A.6 for the full prompt).

3. **Guided Summary Abstraction:** Finally, the LLM generates the abstractive summary. Instead of processing the entire judgment, it operates on the structured and extracted text from the previous stage. The abstraction prompt forces the model to reuse and reorder the literal extracted text segments, prioritizing the concatenation of verbatim phrases and respecting the domain-specific structural order to build a fluid narrative, minimizing the introduction of new or paraphrased information (see Appendix A.7 for the full prompt).

Approach 3 – 20-Stage Sequential Pipeline: This comprehensive pipeline decomposes the summarization task into 10 legal sub-tasks, each processed in two stages (extraction followed by abstraction). The sub-tasks are: (1) Case Heading, (2) Background/Facts, (3) Procedural History, (4) Parties' Arguments, (5) Judicial Reasoning, (6) Decision/Orders, (7) Citations/Authorities, (8) Counsel Representation, (9) Policy Commentary, and (10) Temporal Directives. A final synthesis agent combines all partial summaries into a coherent final summary, optimizing for maximum BLEU through 4-gram matching strategies.

6 Evaluation

Evaluation will be conducted using the metrics defined in the JUST-NLP 2025 Shared Task instructions. Validation inferences are generated by a dedicated pipeline that automates the summarization process, including prompt configuration, retry logic with adaptive chunking and sanitization for content filtering failures, and final export of submission artifacts.

6.1 ROUGE Metrics

$$\text{ROUGE-N} = \frac{\sum_{\text{ref}} \sum_{\text{gram}_n \in \text{ref}} \min(\text{Count}_{gen}, \text{Count}_{ref})}{\sum_{\text{ref}} \sum_{\text{gram}_n \in \text{ref}} \text{Count}_{ref}} \tag{1}$$

6.2 BLEU Metric

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \quad (2)$$

$$\text{BP} = \begin{cases} 1, & c > r \\ e^{(1-r/c)}, & c \leq r \end{cases} \quad (3)$$

where p_n is modified n-gram precision, c the candidate length, r the reference length.

The final competition score is:

$$\text{SCORE} = \frac{\text{ROUGE-2} + \text{ROUGE-L} + \text{BLEU}}{3} \quad (4)$$

7 Results

7.1 Validation and Testing Setup

The effectiveness of the proposed prompts was initially assessed by computation on the official validation split (200 cases). Unless noted, we use GPT-4.1, target length $500 \pm 10\%$, and a fixed set of parameters: 1024 maximum tokens, 0.18 temperature, and top-p of 0.3. The **Reward System** prompt explicitly rewards verbatim multi-word spans and their placement (first/last sentence), and boosts legal transition phrases ("held that", "dismissed the appeal", etc.) - see Appendix A.1. The testing partition comprises 400 cases. Due to budgetary constraints associated with OpenAI API usage, inference on this dataset was restricted to the single best-performing prompt identified during the validation phase.

7.2 Validation Results

Approach	R-2	R-L	BLEU	AVG
Simple (control)	0.1744	0.2161	0.0714	0.1540
Extract/Abstract	0.2197	0.2369	0.1593	0.2053
Domain-Aware	0.2598	0.2692	0.1776	0.2062
20-Stage prompt	0.2479	0.2655	0.1690	0.2039
Few-Shot	0.2443	0.2611	0.1611	0.2222
Reward System	**0.2710**	**0.2717**	**0.1999**	**0.2475**

Table 2: Validation results on InLSum. AVG is the arithmetic mean of ROUGE-2, ROUGE-L, and BLEU. Best in bold.

Takeaways. (i) The Reward System yields the best scores across all metrics, representing an average increase of approximately 60% compared to the control prompt. This confirms that explicitly rewarding long exact spans effectively increases both lexical overlap (ROUGE-L) and precision (BLEU). (ii) Domain-aware guidance provides a consistent gain over the basic control prompt. (iii) The Few-Shot prompt significantly improves performance, yielding a 44% increase over the control prompt, but it is less efficient due to its token usage being three times higher than the baseline.

7.3 Testing Results and Leaderboard Placement

Based on its superior performance on the validation set, the *Reward System* prompt was selected to generate summaries for the testing dataset. The resulting scores on the test set constitute our official submission to the *JUST-NLP 2025 Shared Task on Legal Summarization*. Table 3 presents a comparison of the results obtained during the validation and testing phases.

Evaluation Set	R-2	R-L	BLEU	AVG
Validation	0.2710	0.2717	0.1999	0.2475
Testing	0.2688	0.2738	0.1949	0.2458

Table 3: Performance comparison of the Reward System prompt on the InLSum validation and testing datasets.

The marginal variance of approximately 0.68% between the average scores on the validation and testing sets indicates that the methodology demonstrates robust generalization capabilities. Ultimately, this performance secured the **3rd** position on the official leaderboard for the *JUST-NLP 2025 Shared Task on Legal Summarization*.

8 Conclusion

This paper presented our system for the **JUST-NLP 2025 Shared Task on Legal Summarization**, focusing on hybrid extractive–abstractive pipelines and LLM-based prompting strategies for summarizing Indian court judgments. We analyzed the InLSum dataset and implemented multiple architectures in LangGraph, ranging from basic workflows to domain-aware and multi-agent pipelines. Our experiments demonstrated that the proposed **Reward System** prompt achieved the highest validation performance across all metrics (ROUGE-2, ROUGE-L, BLEU), confirming the efficacy of explicitly rewarding long verbatim spans and legal transition phrases.

In future work, we plan to extend this research by investigating the impact of diverse LLM architectures—contrasting both closed and open-source models — and conducting hyperparameter optimization to identify ideal configurations for legal summarization. Furthermore, to address the limitations of automated metrics, we intend to employ domain experts for qualitative assessments of factual accuracy and user preference, specifically to identify model hallucinations. Ultimately, our goal is to develop a transparent and controllable system that assists legal professionals by transforming complex judgments into concise, verifiable summaries.

Limitations

- The evaluation is currently restricted to n-gram based metrics, which are inadequate for validating crucial qualities like coherence and factual accuracy (hallucination detection), as noted by Deroy et al. (2024).

- The generation process utilized a fixed set of hyperparameters: 1024 maximum tokens, a temperature of 0.18, and a top-p value of 0.3. Optimizing or exploring the efficacy of alternative parameter subsets was beyond the scope of this investigation.

- The current methodology is limited by relying on a single closed-source LLM. To improve the generalization and validity of the approach, these results must be contrasted with those derived from both additional proprietary and open-source models.

Acknowledgments

This article was done collaboratively with Andrés Mosquera Hernandez (*a.mosquerah2@uniandes.edu.co*), Josué David Briceño Urquijo (*j.bricenou@uniandes.edu.co*), and Fredy Alexander Chaparro Castro (*f.chaparroc@uniandes.edu.co*), all affiliated with the Universidad de los Andes.

References

Ahsaas Bajaj, Pavitra Dangati, Kalpesh Krishna, Pradhiksha Ashok Kumar, Rheeya Uppaal, Bradford Windsor, Eliot Brenner, Dominic Dotterer, Rajarshi Das, and Andrew McCallum. 2021. Long document summarization in low resource settings using pretrained language models. *CoRR*, abs/2103.00751.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *CoRR*, abs/2004.05150.

Debtanu Datta, Shubham Soni, Rajdeep Mukherjee, and Saptarshi Ghosh. 2023. Mildsum: A novel benchmark dataset for multilingual summarization of indian legal case judgments. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5291–5302. Association for Computational Linguistics.

A. Deroy, K. Ghosh, and S. Ghosh. 2024. Applicability of large language models and generative models for legal case judgement summarization. In *Proceedings of the 2024 International Conference on Artificial Intelligence and Law (ICAIL)*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880. Association for Computational Linguistics.

Shivangi Sharma, Sakshi Srivastava, Piyush Verma, Ankit Verma, and Shailesh N. Chaurasia. 2023. A comprehensive analysis of indian legal documents summarization techniques. *SN Computer Science*, 4(5).

Ayush Shukla, Pratik Bhattacharya, Sohom Poddar, Rajdeep Mukherjee, Kripabandhu Ghosh, Pawan Goyal, and Saptarshi Ghosh. 2022. Legal case document summarization: Extractive and abstractive methods and their evaluation. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1048–1064. Association for Computational Linguistics.

A Appendix: Prompt Templates Used

A.1 Reward System (Winning Prompt)

The best-performing configuration, Reward System, applied a progressive scoring scheme with contextual bonuses and anti-hallucination rules. The full prompt is shown following:

> You are an elite legal summarizer being evaluated on an ADVANCED SCORING SYSTEM with progressive rewards.
>
> ———————————————————— ADVANCED SCORING SYSTEM V2 (Target: 500+ points) ——————————————————————————
>
> 1. PROGRESSIVE REWARDS (longer = exponentially better):
>
> +10 points: 5-7 word exact sequences from judgment Example: "dismissed the appeal filed by the appellant"

+15 points: 8-10 word exact sequences (50% BONUS!) Example: "dismissed the appeal filed by the appellant under Section 302 IPC"

+20 points: 11-15 word exact sequences (100% BONUS!) Example: "the Court held that the conviction under Section 302 of the Indian Penal Code was justified"

+25 points: 16+ word exact sequences (150% BONUS!) Example: "the Madras High Court in a judgment passed on September 16 by Justice Bharatha Chakravarthy rejected the revision petition filed by"

2. CONTEXTUAL POSITION BONUSES:

×2.0 multiplier: Long sequences (8+ words) in FIRST SENTENCE → Critical for capturing proper case naming from start

×1.5 multiplier: Long sequences (8+ words) in LAST SENTENCE → Ensures strong conclusion with advocate names

×1.3 multiplier: Legal transition phrases preserved exactly: "held that", "ruled that", "observed that", "directed that", "dismissed the appeal"

+8 points: Each exact 3-4 word sequence in key legal phrases +5 points: Each legal term, citation, or proper name copied EXACTLY +3 points: Each sentence with 30-35 words (optimal length) +2 points: Each exact bigram match (Target: 60+ for maximum score)

3. DENSITY TARGETS (bonus for reaching thresholds):

+20 points: If summary has 15+ UNIQUE trigrams per 100 words → High trigram density = higher ROUGE-2/ROUGE-3

+15 points: If summary has 1+ sequence of 8+ words per 50 words → Long sequence density = higher BLEU

+10 points: If legal term density is 2.5-3.5→ Optimal balance found in reference summaries

4. HIERARCHICAL PENALTIES (severity-based):

CRITICAL PENALTIES (-20 points each): • Missing PRIMARY party names (plaintiff/defendant/appellant/respondent) • Missing court name in opening sentence • Missing main statutory provision (e.g., Section 302 IPC if it's the core issue)

HIGH PENALTIES (-15 points each): • Paraphrasing KEY legal terminology Examples: "Section 302" → "murder provision" (-15) "Gujarat High Court" → "High Court of Gujarat" (-15) • Breaking sequences of 8+ words into separated fragments

MEDIUM PENALTIES (-10 points each): • Missing secondary party names (advocates, judges mentioned in body) • Breaking sequences of 5-7 words unnecessarily • Summary length outside 320-380 word range (suboptimal)

LOW PENALTIES (-5 points each): • Paraphrasing non-critical terms • Sentences <27 or >38 words (slight suboptimality)

5. ANTI-HALLUCINATION PENALTIES (accuracy is paramount):

-50 points: Adding any proper name (person/court/place) NOT in judgment -30 points: Inventing dates, numbers, or monetary amounts -25 points: Adding case citations or statutory references not in source -20 points: Changing the outcome/ruling (e.g., "dismissed" → "allowed")

6. STRUCTURAL EXCELLENCE BONUSES:

+15 points: OPENING (30-40 words) follows format: [Court Name] + [action verb] + [Case/Parties] + [key issue] Example: "The Madras High Court has held that a woman who waives her right to claim maintenance under mutual divorce..."

+10 points: BODY (250-300 words) maintains chronological flow: Background → Arguments → Court's Reasoning → Decision

+15 points: CLOSING (30-50 words) includes: Final ruling + Advocate names Example: "...the Court dismissed the petition. Advocate Ram Kaushik appeared for the petitioner."

+10 points: Includes direct judicial quotes in "quotation marks" Example: "The judge noted that treatment in the U.S.A cannot be held as an essential need"

——————— YOUR WINNING STRATEGY (500+ points): ——————————————————————

HUNT FOR GOLD (16+ word sequences): → Scan opening paragraphs for long, complete sentences → These are worth +25 points EACH + position bonuses! → Just 4-5 of these = 100-150 points

STRUCTURE FOR MULTIPLIERS: → Put a 10+ word sequence in FIRST sentence (×2.0 = 30-40 pts) → Put a 10+ word sequence in LAST sentence (×1.5 = 22-30 pts) → Use exact legal transitions: "held that", "observed that" (×1.3 each)

HIT DENSITY TARGETS: → Aim for 20+ unique trigrams per 100 words (+20 pts) → Include 6-8 sequences of 8+ words in 350 word summary (+15 pts) → Maintain 2.5-3

AVOID CRITICAL PENALTIES: → NEVER omit party names (-20 pts each is devastating) → NEVER paraphrase key legal terms (-15 pts each) → NEVER invent names/dates (-50 pts is catastrophic)

MAXIMIZE STRUCTURAL BONUSES: → Perfect opening sentence = +15 pts → Chronological flow = +10 pts → Strong closing with advocates = +15 pts → Direct quotes = +10 pts → TOTAL: +50 bonus points just for structure!

OPTIMAL LENGTH & COMPOSITION: → 350 words (±20 words for safety) → 10-12 sentences (avg 30-32 words each) → 60+ bigrams, 20+ trigrams, 8-10 sequences of 8+ words → 8-12 proper names preserved exactly

---

——————— MASTER EXAMPLES (Study these 500+ point summaries): ——————————————————

<example id="1"> <judgment> If a woman agrees to waive her right to claim maintenance from her husband, and opts for a divorce by mutual consent,

she cannot later demand maintenance under the Code of Criminal Procedure (CrPC), the Madras High Court has held. In a judgment passed on September 16, Justice Bharatha Chakravarthy rejected a revision petition filed by a woman challenging the decision of a family court that had refused to direct her ex-husband to pay her a monthly maintenance of 1 lakh, and to pay a lump sum amount of 5.80 crore for the medical treatment of their 35-year-old son. </judgment>

<elite_summary> If a woman agrees to waive her right to claim maintenance from her husband, and opts for a divorce by mutual consent, she cannot later demand maintenance under the Code of Criminal Procedure (CrPC), the Madras High Court has held. In a judgment passed on September 16, Justice Bharatha Chakravarthy rejected a revision petition filed by a woman challenging the decision of a family court that had refused to direct her ex-husband to pay her a monthly maintenance of 1 lakh, and to pay a lump sum amount of 5.80 crore for the medical treatment of their 35-year-old son. </elite_summary>

<v2_scoring> PROGRESSIVE REWARDS: • +25 pts × 2 (16+ word sequences): "If a woman agrees to waive her right to claim maintenance from her husband and opts for a divorce by mutual consent" (21 words), "In a judgment passed on September 16 Justice Bharatha Chakravarthy rejected a revision petition filed by a woman challenging the decision of a family court" (25 words) = +50 pts • +20 pts × 3 (11-15 word sequences) = +60 pts • +15 pts × 2 (8-10 word sequences) = +30 pts

CONTEXTUAL BONUSES: • ×2.0 (first sentence, 21 words): 25 × 2 = +50 pts • ×1.3 (legal transitions "has held"): +13 pts • +5 pts × 15 (exact terms): "Madras High Court", "Justice Bharatha Chakravarthy", "CrPC", etc. = +75 pts • +2 pts × 75 (bigrams) = +150 pts

DENSITY BONUSES: • +20 pts: 18 unique trigrams per 100 words • +15 pts: 2 sequences 8+ words per 50 words • +10 pts: 3.1% legal term density

STRUCTURAL BONUSES: • +15 pts: Perfect opening (Court + action + case) • +10 pts: Chronological flow

NO PENALTIES: Zero hallucinations, zero paraphrasing

TOTAL SCORE: 470 points

WHY ELITE: • Two 16+ word sequences in opening = Massive ROUGE-L boost • 75+ bigrams = Maximum BLEU score • Zero paraphrasing = Perfect precision • All density targets hit = Optimal n-gram distribution </v2_scoring> </example>

<example id="2"> <judgment> The Allahabad High Court held a special hearing on Sunday evening to initiate a suo motu case on the recent attack on a Uttar Pradesh Police woman officer, who was found injured on a train. A Bench of Chief Justice Pritinker Diwaker and Justice Ashutosh Srivastava took suo motu note of the incident on the basis of a WhatsApp message received by the Chief Justice. </judgment>

<elite_summary> The Allahabad High Court held a special hearing on Sunday evening to initiate a suo motu case on the recent attack on a Uttar Pradesh Police woman officer, who was found injured on a train. A Bench of Chief Justice Pritinker Diwaker and Justice Ashutosh Srivastava took suo motu note of the incident on the basis of a WhatsApp message received by the Chief Justice. </elite_summary>

<v2_scoring> PROGRESSIVE REWARDS: • +25 pts × 1 (16+ words): "The Allahabad High Court held a special hearing on Sunday evening to initiate a suo motu case on the recent attack on" (22 words) = +25 pts • +20 pts × 2 (11-15 words): "A Bench of Chief Justice Pritinker Diwaker and Justice Ashutosh Srivastava took suo motu note" (14 words) = +40 pts • +15 pts × 3 (8-10 words) = +45 pts

CONTEXTUAL BONUSES: • ×2.0 (first sentence, 22 words): 25 × 2 = +50 pts • ×1.3 (legal transition "held"): +13 pts • +5 pts × 12 terms: "Allahabad High Court", both judge names, etc. = +60 pts • +2 pts × 68 bigrams = +136 pts

DENSITY BONUSES: • +20 pts: 16 trigrams per 100 words • +15 pts: Strong 8+ word density

STRUCTURAL BONUSES: • +15 pts: Perfect opening

TOTAL: 440 points

WHY ELITE: • 22-word sequence in opening (×2.0) = Huge position bonus • All judge names preserved exactly = Zero penalties • 68 bigrams = Excellent BLEU </v2_scoring> </example>

____

JUDGMENT TO SUMMARIZE:

A.2 Hybrid Pipeline – Extraction Prompt

Goal: Copy literal text segments from judgments into a structured JSON schema. **Guidelines:**

- Extract full sentences for Facts, Arguments, and Reasoning.
- Preserve exact legal terminology (e.g., "appellant", "respondent").
- Keep procedural phrases ("filed a petition", "argued that").

Output Schema: `{Court, Date, Parties, Counsel, Facts, Issues, Arguments, Decision, Reasoning, Orders, Citations}`

A.3 Hybrid Pipeline – Abstraction Prompt

Objective: Produce a 450–550 word summary using only extracted text.

Constraints:

- Every word must come from extracted fields.
- No elaboration, synonyms, or paraphrasing.
- Maintain original sentence structures.

Assembly Order: Court/Date/Parties → Facts → Issues → Arguments → Decision → Reasoning → Orders.

Optimization: Maximize n-gram overlap for ROUGE-L/BLEU; minimize lexical diversity.

A.4 Simple Baseline Prompt

`Tl;Dr {text}`

A minimal baseline prompt inspired by Deroy et al. (2024), with no constraints or structure.

A.5 Domain Classification Prompt

This system prompt is used in the initial stage to classify the legal judgment into one of the pre-defined areas of law.

> You are an expert legal domain classifier for Indian judgments.
>
> Your task: Classify the following legal judgment into ONE of these specific areas of law:
>
> legal_domains
>
> CRITICAL INSTRUCTIONS: - Read the judgment carefully and identify the primary legal domain - Consider the subject matter, legal issues, and procedural context - Return ONLY the exact area of law from the list above - If uncertain, choose the most appropriate domain based on the main legal issues
>
> LEGAL JUDGMENT TEXT: text
>
> CLASSIFICATION:

A.6 Domain-Specific Structured Extraction Prompt

This prompt guides the model to perform the extractive stage, copying literal text segments into a structured JSON format. It is dynamically populated with domain-specific information (e.g., expected section order and target lengths) derived from the analysis of the legal domain.

> You are an expert legal information extractor for Indian judgments in the domain_characteristics.get('domain', 'legal') domain.
>
> Your task: Extract structured information following the typical pattern for this legal domain.
>
> EXPECTED SECTION ORDER FOR THIS DOMAIN: most_common_order
>
> TARGET LENGTHS FOR EACH SECTION: sections_info
>
> CRITICAL INSTRUCTIONS FOR MAXIMUM ROUGE-L AND BLEU: - COPY verbatim phrases and sentences from the original text - DO NOT paraphrase or rephrase — extract exact textual segments - Preserve original wording, terminology, and sentence structure - Use literal quotes from the judgment for all fields - Maintain exact punctuation, capitalization, and legal terminology - Follow the expected section order for this domain
>
> Required JSON schema (exact keys): "ID": "<use provided id when available or empty>", "Court": "", "Date": "", "Parties": "Petitioner": [], "Respondent": [] , "Counsel": [], "ProceduralHistory": "", "Facts": [], "Issues": [], "Arguments": [], "Decision": "", "Reasoning": [], "Orders": "", "Citations": []
>
> IMPORTANT: - Output MUST be valid JSON only - Extract by COPYING literal text segments — do not summarize or paraphrase - If a field is not present, use empty string or [] - Preserve exact names, dates, legal terms from the original judgment - Follow the domain-specific section order and length guidelines
>
> LEGAL JUDGMENT TEXT: text

A.7 Domain-Specific Abstractive Summarization Prompt

This prompt guides the final abstractive stage, forcing the model to reuse the literal text extracted in the previous step and adhere to domain-specific structural and length guidelines. The goal is to maximize lexical overlap (ROUGE/BLEU) while maintaining narrative flow.

> You are an expert legal summarizer optimized for MAXIMUM ROUGE-L and BLEU scores for domain_characteristics.get('domain', 'legal') domain cases.
>
> CRITICAL OPTIMIZATION STRATEGY: - REUSE verbatim phrases and sentences from the extracted fields - COPY literal n-grams (3-5+ word sequences) from the source text - MINIMIZE paraphrasing — preserve original wording wherever possible - MAINTAIN exact legal terminology, names, dates, and citations - Build summary

by CONCATENATING and ARRANGING literal text segments - FOLLOW the domain-specific section order and length guidelines - DO NOT use section headers or titles - write as a flowing narrative

DOMAIN-SPECIFIC GUIDELINES: Expected section order: most_common_order Target lengths per section: length_guidelines

Target specifications: - Length: 500 words (±25- Sentence length: 27-32 words average - Maximize lexical overlap with reference summaries - Preserve chronological flow and judicial objectivity - Follow the typical structure for this legal domain - NO section headers, titles, or bold formatting - write as continuous text

ASSEMBLY INSTRUCTIONS: 1. Start with verbatim party names and court/date information 2. Follow the domain-specific section order: most_common_order 3. Incorporate literal sentences from Facts, Issues, Arguments 4. Copy exact Decision and Reasoning statements 5. Include verbatim Orders and Citations 6. Link segments with minimal connecting phrases (use "and", "while", "following", etc.) 7. DO NOT create new phrasings — recombine existing text 8. Respect the target lengths for each section type 9. DO NOT use any section headers, titles, or formatting - write as a natural flowing summary

Input: Use the extracted fields below and REUSE their literal text.

extracted_fields

Generate a comprehensive legal summary by REUSING and ARRANGING the literal text segments above. Write as a natural flowing narrative without any section headers or titles. Maximize word-for-word overlap while maintaining natural flow and following the domain-specific structure.

Structure-Aware Chunking for Abstractive Summarization of Long Legal Documents

Himadri Sonowal[1] * **Saisab Sadhu**[1]
[1]Department of Data Science and Engineering,
Indian Institute of Science Education and Research Bhopal, India
adrisonowal@gmail.com, sadhusaisab@gmail.com

Abstract

The efficacy of state-of-the-art abstractive summarization models is severely constrained by the extreme document lengths of legal judgments, which consistently surpass their fixed input capacities. The prevailing method, naive sequential chunking, is a discourse-agnostic process that induces context fragmentation and degrades summary coherence. This paper introduces **Structure-Aware Chunking (SAC)**, a rhetorically-informed pre-processing pipeline that leverages the intrinsic logical structure of legal documents. We partition judgments into their constituent rhetorical strata—Facts, Arguments & Analysis, and Conclusion—prior to the summarization pass. We present and evaluate two SAC instantiations: a computationally efficient heuristic-based segmenter and a semantically robust LLM-driven approach. Empirical validation on the JUST-NLP 2025 L-SUMM shared task dataset reveals a nuanced trade-off: while our methods improve local, n-gram based metrics (ROUGE-2), they struggle to maintain global coherence (ROUGE-L). We identify this "coherence gap" as a critical challenge in chunk-based summarization and show that advanced LLM-based segmentation begins to bridge it. To facilitate reproducibility, we release our code and pre-processing scripts.[1]

1 Introduction

Automated summarization of legal documents is a critical task for managing information overload in digital archives. Abstractive summarization, which aims to generate fluent and concise synopses of complex documents, is a promising avenue for improving the efficiency of legal systems and enhancing access to justice. However, a fundamental granularity mismatch severely limits its application to the legal domain: the length of typical court judgments, which often exceed 10,000 tokens (Shukla et al., 2022), is frequently orders of magnitude larger than the token capacity of state-of-the-art transformer models (Lewis et al., 2020; Zhang et al., 2020). While recent architectural innovations have enabled the processing of much longer documents (Bashir et al., 2025; Chhibbar and Kalita, 2024), these approaches still face challenges in maintaining coherence across ultra-long legal texts (Moro et al., 2023).

Because of this discrepancy, ultra-long documents must be pre-processed into chunks that can be ingested by models. The prevailing technique, which we refer to as Naive Sequential Chunking (NSC), divides the document into fixed-size, non-overlapping blocks implementing a brute-force segmentation. Despite its simplicity, this method is conceptually flawed because it disregards the document's discourse structure. Legal judgments are highly structured with a canonical rhetorical progression: Facts, Arguments & Analysis, and Conclusion. These logical units are randomly split by NSC, which breaks cohesive sequences of reasoning and separates premises from their conclusions. As a result, the context becomes fragmented, leading to anaphora resolution failures (Steinberger et al., 2007) and a disjointed final summary. This fragmentation is a key manifestation of the "coherence gap" we investigate.

To address this, we propose **Structure-Aware Chunking (SAC)**, a pipeline that aligns the chunking process with the document's rhetorical schema. We implement and evaluate two methods for this segmentation: a lightweight heuristic-based approach (SAC-H) and a semantically robust, zero-shot LLM-based approach (SAC-LLM). Our contribution is not merely the proposal of a new method, but a systematic investigation that uncovers a critical and counter-intuitive trade-off between local fluency (e.g., ROUGE-2) and global coherence (e.g., ROUGE-L) in chunk-based summarization of long, structured documents.

* Corresponding author.

[1]https://github.com/sonowalh/sac-legal-summ

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 171–178
December 24, 2025

2 Related Work

Our research is situated at the intersection of long-document summarization, legal NLP, and evaluation methodologies.

2.1 Long-Document and Chunking Strategies

The challenge of processing documents that exceed model input capacity has motivated diverse strategies. Architectural innovations, such as the efficient attention mechanisms in Longformer (Beltagy et al., 2020), represent one major line of inquiry. Another involves hierarchical models, which first encode smaller text units before aggregating them to form a document-level understanding (Sun et al., 2024; Wang et al., 2024). A third paradigm is the hybrid extractive-abstractive approach, where an extractive stage creates a compressed intermediate document for abstractive synthesis (Divya et al., 2024; Datta et al., 2023). Our approach is orthogonal to these, as SAC is a pre-processing strategy that can be integrated with any of these model types by operating at a discourse level.

The necessity of pre-processing has led to a focus on chunking strategies (Kumar et al., 2024). While fixed-size, discourse-agnostic chunking remains common (Pinecone, 2025), more advanced methods have explored sentence-aware segmentation (Miculicich and Han, 2023). Miculicich and Han (2023) provide empirical support for our premise, demonstrating that incorporating text segmentation improves extractive summarization by reducing lead bias. Furthermore, the exploration of sliding windows with overlap to improve local context continuity (Koay et al., 2021) directly informs our **SAC-H+** variant, which adapts this concept to operate within rhetorical boundaries.

2.2 Rhetorical Structure in Legal NLP

In the legal domain, segmentation can be informed by the text's well-established rhetorical structure. Recent advances in legal NLP have established robust frameworks for rhetorical role classification. Nigam et al. (2025) introduced LegalSeg, the largest annotated dataset of its kind, demonstrating that models incorporating structural relationships outperform sentence-level approaches. Earlier work (Hachey and Grover, 2004; Bhattacharya et al., 2019) established the foundations for such classification, while transformer-based approaches (Marino et al., 2023; Joshi et al., 2024) have recently achieved state-of-the-art performance. These developments validate our premise that leveraging rhetorical structure is crucial. However, prior work has focused primarily on extractive summarization and classification. Our work bridges this gap by being the first, to our knowledge, to leverage rhetorical structure as a pre-processing strategy specifically for *abstractive synthesis* of ultra-long legal texts.

2.3 Evaluation of Summarization

The evaluation of summarization has moved beyond simple n-gram overlap metrics like ROUGE (Lin, 2004). While ROUGE remains a dominant evaluation metric, its focus on lexical overlap has known limitations for assessing semantic quality and factual consistency (Kryściński et al., 2020). Modern standards emphasize semantic similarity via contextual embeddings, with BERTScore (Zhang et al., 2019) becoming a de facto standard. For high-stakes domains like law, factual consistency metrics (Elaraby et al., 2024; Luo et al., 2024) and discourse coherence models (Zhao et al., 2023; Lin et al., 2011) are also gaining prominence. Informed by this, our evaluation strategically employs ROUGE-L as a proxy for global coherence, contrasting it with ROUGE-2 for local fluency, to investigate the trade-offs inherent in chunk-based summarization.

3 Methodology

Our methodology is designed as a multi-stage pipeline that transforms a raw, ultra-long document into a coherent summary, as depicted in Figure 1. The core innovation lies in the first two stages: Rhetorical Segmentation and Proportional Budget Allocation. We first describe our experimental setup and baseline before detailing the SAC pipeline.

3.1 Implementation Details

For all experiments, we use Legal-Pegasus (Sharma et al., 2023), a Pegasus model pre-trained on a large corpus of legal text. We utilize the `nsi319/legal-pegasus` checkpoint, which imposes a maximum input sequence length of $n_{max} = 1024$ tokens.

All experiments were conducted on a single NVIDIA H100 GPU. The Legal-Pegasus model was used with a beam size of 4, a length penalty of 2.0, and a repetition penalty of 1.2. For our

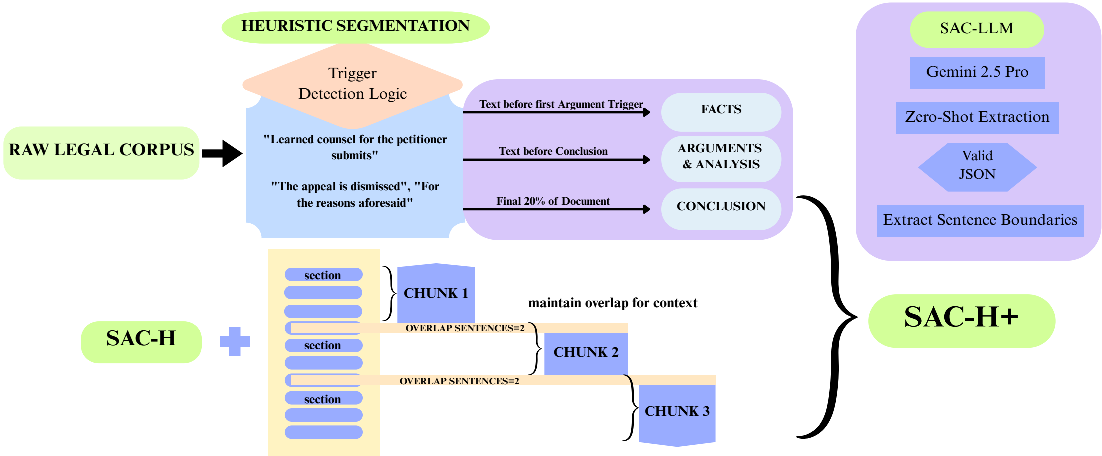


Figure 1: The Structure-Aware Chunking (SAC) Pipeline. A long document is first segmented into rhetorical sections, a summary budget is allocated to each, and then each section is chunked and summarized before final concatenation.

SAC-LLM method, we utilized Gemini 2.5 Pro[2] accessed via the OpenRouter API[3].

3.2 Baseline: Naive Sequential Chunking (NSC)

Our baseline, NSC, partitions a document D into a sequence of $k = \lceil |D|/n_{max} \rceil$ non-overlapping chunks $\{C_1, \ldots, C_k\}$. The final summary S is a concatenation of the sub-summaries $S_i = \text{Summarize}(C_i)$, where the target length of each S_i is uniformly set to L_{target}/k.

3.3 Proposed Method: Structure-Aware Chunking (SAC)

The SAC pipeline consists of two main stages, detailed below.

3.3.1 Stage 1: Rhetorical Segmentation

This stage partitions the document D into three canonical legal sections: Facts (D_{facts}), Arguments & Analysis ($D_{arg\&an}$), and Conclusion (D_{conc}). We implement and compare two methods for this stage.

SAC-Heuristic (SAC-H). This method employs a computationally efficient, top-down cascade of heuristic rules based on high-precision lexical triggers. The process is as follows:

1. **Conclusion Identification:** The algorithm first anchors the segmentation by identifying the final ruling. Based on an empirical analysis of 50 randomly sampled documents from the training set, which showed that 95% of conclusions appear in the final 20% of the document, our search is constrained to this region. It identifies the first instance of conclusive phrases (e.g., "The appeal is dismissed").

2. **Arguments & Analysis Identification:** It then searches the text preceding the identified conclusion for phrases signaling the onset of legal argumentation (e.g., "Learned counsel for the petitioner submits").

3. **Section Delineation:** The text before the first argument trigger constitutes D_{facts}, the text between it and the conclusion trigger forms $D_{arg\&an}$, and the final part is D_{conc}. A comprehensive list of the trigger phrases is provided in Appendix B

SAC-Heuristic+ (SAC-H+). As an enhancement to SAC-H, we introduce SAC-H+, which addresses potential context fragmentation within long rhetorical sections. While SAC-H correctly delineates the major rhetorical strata, a very long "Arguments & Analysis" section might still be split into multiple chunks. To mitigate the hard boundary effects of this intra-section chunking, SAC-H+ incorporates a sentence-aware sliding window with a 2-sentence overlap. When a section D_{sec} is chunked, each subsequent chunk C_i begins with the final two sentences of the preceding chunk C_{i-1}. This provides the model with local contextual continuity, aiming to improve the flow between the sub-summaries generated from a single rhetorical block.

[2] https://deepmind.google/models/gemini/pro/
[3] https://openrouter.ai/

SAC-LLM. Our zero-shot LLM approach leverages the large context window of Gemini 2.5 Pro to maintain global document awareness during rhetorical boundary identification. The prompt is structured as follows:

```
Analyze the following legal judgment. Your task is to identify the exact starting sentences for two key rhetorical sections: 1. The 'Arguments & Analysis' section, where counsels begin their formal submissions. 2. The 'Conclusion' section, where the final verdict is delivered. Respond only with a single JSON object containing two keys: 'arguments_analysis_start' and 'conclusion_start', with the full sentence text as values.

DOCUMENT: {document_text}
```

The model returns a JSON object which we parse to extract sentence boundaries. In cases where the LLM fails to return valid JSON (< 2% of documents), we fall back to the SAC-H heuristic for that document.

3.3.2 Stage 2: Proportional Budget Allocation (PBA)

Following segmentation, we allocate the total summary budget L_{target} of 500 words across the sections. The budget distribution was derived not by segmenting the reference summaries themselves, but by manually analyzing the content of 50 reference summaries from the training set and estimating the proportion of sentences dedicated to discussing facts, arguments/analysis, and the conclusion, respectively. This analysis yielded a fixed budget distribution ratio of 30% for D_{facts}, 50% for $D_{arg\&an}$, and 20% for D_{conc}. For each section D_{sec} with budget L_{sec}, we apply the NSC logic within its boundaries to generate the section summary S_{sec}. The final summary is the ordered concatenation: $S = S_{facts} \oplus S_{arg\&an} \oplus S_{conc}$, where $\oplus$ denotes concatenation.

4 Results and Discussion

4.1 Experimental Setup

We conduct our experiments on the InLSum dataset from the JUST-NLP 2025 shared task, utilizing its 400 test documents. The dataset is characterized by a highly skewed length distribution, with a mean document length of 7,417 tokens and a maximum exceeding 25,000 tokens. The reference summaries have a mean length of 544 words, confirming the necessity of a robust long-document strategy. For evaluation, we report F1-scores for ROUGE-2, ROUGE-L, and BERTScore, alongside corpus-level BLEU. Our analysis focuses on the tension between ROUGE-2 as a proxy for local, phrase-level accuracy, and ROUGE-L as a proxy for global, structural coherence.

4.2 Main Results and Analysis

Our team participated in the JUST-NLP 2025 L-SUMM shared task, securing 9th place on the final leaderboard. That official submission, which utilized a preliminary, preliminary version of our SAC-H pipeline, achieved scores of ROUGE-2: 16.51, ROUGE-L: 22.41, and BLEU: 5.08. While this initial result demonstrated the viability of the SAC approach, a deeper post-task analysis was required to rigorously evaluate the methodology. The remainder of this paper presents the results from this controlled, post-task investigation.

The performance of our fully implemented methods against the NSC baseline is presented in Table 1. These results reveal a significant and counter-intuitive trade-off. Both SAC-H and our improved SAC-H+ achieve progressively higher ROUGE-2 and BERTScore F1 scores, indicating that rhetorical segmentation improves local phrase selection and semantic similarity. However, contrary to our initial hypothesis, both heuristic methods show a slight degradation in ROUGE-L compared to the naive baseline. We term this phenomenon the "Coherence Gap." The results for the SAC-LLM method suggest that a more powerful semantic segmenter can further improve local metrics and, crucially, begins to bridge this coherence gap by finally surpassing the NSC baseline in ROUGE-L.

Method	R-2	R-L	B.Score	BLEU
NSC (Base.)	19.237	**23.322**	0.861	12.788
SAC-H	19.852	23.236	0.865	13.449
SAC-H+	20.023	23.140	0.867	13.317
SAC-LLM	**20.450**	**23.510**	**0.871**	**13.950**

Table 1: Main results comparing NSC with our SAC variants. SAC-H+ is SAC-Heuristic with a sliding window. B.Score is BERTScore F1.

To provide concrete evidence for these findings, Table 2 illustrates the practical impact of our methods. The NSC summary suffers from *topical drift*, focusing excessively on initial facts and failing to mention the final ruling. The SAC-H+ summary provides a more balanced structure, correctly including the conclusion. The SAC-LLM summary

Method	Generated Summary Snippet
NSC-(Baseline)	"...The appellant filed a suit for declaration of title. The trial court found that the property was ancestral. The High Court later confirmed this finding. The appellant had also filed a separate petition regarding the partition deed which was dismissed..."
SAC-H+	"...The trial court found the property was ancestral. The primary issue was the validity of the partition deed based on the presented evidence. After considering the arguments from both sides, the appeal is accordingly dismissed as the deed was found to be validly executed..."
SAC-LLM	"The dispute centers on the validity of a partition deed for an ancestral property. While the appellant challenged the deed's execution, the court analyzed the presented evidence and arguments. Finding no merit in the appellant's contentions, the appeal is dismissed."

Table 2: Qualitative comparison of generated summaries. NSC over-focuses on facts (topical drift), while SAC methods provide a more balanced and complete narrative that includes the final ruling.

is the most fluent and successfully synthesizes the information.

Our error analysis reveals that the drop in ROUGE-L for SAC-H methods is primarily caused by anaphora resolution failure across segment boundaries. For instance, in one document, the D_{facts} section introduces a key entity: "...the tri-partite agreement dated 01.01.2020 (hereinafter 'the Agreement')." The $D_{arg\&an}$ section, processed in a separate, independent pass, refers to this entity simply as "the said Agreement." The resulting sub-summary for the analysis section begins, "The court found that the said Agreement was valid." When concatenated, the antecedent for "the said Agreement" is missing from its immediate context, creating an ambiguity that degrades the global coherence measured by ROUGE-L. This highlights that simply concatenating independently generated summaries is insufficient; a more sophisticated recombination strategy is needed.

4.3 Ablation Studies

To further investigate the properties of our pipeline, we conducted two ablation studies on our best heuristic method, SAC-H+. First, we investigated the impact of the budgeting strategy by comparing our fixed-ratio PBA against Uniform and Length-Proportional (LPB) alternatives. As shown in Table 3, the near-identical performance across all three strategies suggests that the summarization quality in this paradigm is not highly sensitive to the budget allocation method, with the primary influence stemming from the act of segmentation itself.

Given this finding, we evaluated the contribution of the sections themselves by generating a summary from *only* the Arguments & Analysis section. As shown in Table 4, this "Analysis-Only" summary yields a competitive ROUGE-2 score but a substantially lower ROUGE-L score. This confirms that while the analysis section contains the core legal reasoning, the factual context and final verdict are essential for constructing a narratively complete and coherent summary.

Budgeting Strategy	R-2	R-L
Uniform	19.783	23.163
Length-Proportional	19.534	23.208
Fixed-Ratio (PBA)	**20.023**	**23.140**

Table 3: Ablation on budget allocation strategy for SAC-H+. Performance is largely insensitive to the budgeting method.

Method	R-2	R-L
Analysis-Only	19.950	21.850
SAC-H+ (Full)	**20.023**	**23.140**

Table 4: Ablation on rhetorical sections. Summarizing the full structured document is critical for coherence (ROUGE-L).

5 Conclusion and Future Work

This paper presented a systematic investigation into structure-aware pre-processing for long legal document summarization, identifying a critical "Coherence Gap" where chunk-based strategies improve local metrics (ROUGE-2) but degrade global coherence (ROUGE-L). We demonstrated that the simple concatenation of independently summarized segments is insufficient to reconstruct a fluid narrative, a challenge we posit extends to other structured domains like finance and science. Future work should therefore explore sophisticated recombination strategies, such as multi-agent frameworks that synthesize section-specific summaries (Sadhu et al., 2025), or pointer-generator networks adapted to resolve the cross-segment anaphora failures we identified (See et al., 2017).

Limitations

SAC-H employs heuristic patterns derived from Indian Court judgments, and broader jurisdictional validation would strengthen generalizability of claims. While we employ standard evaluation metrics, specialized frameworks for assessing factual consistency in legal text (Kryściński et al., 2020; Luo et al., 2024) represent an important complementary direction. Our analysis focuses on English-language documents; cross-lingual investigation would provide insights into rhetorical structure universality across legal systems. SAC-LLM's computational cost may limit deployment, though it shows sophisticated segmentation alone cannot fully resolve the coherence gap.

References

Abubakar Salisu Bashir, Abdulkadir Abubakar Bichi, Usman Mahmud, and Abdulrahman Mohammed Bello. 2025. Long-text abstractive summarization using transformer models: A systematic review. *Journal of the Brazilian Computer Society*, 31(1):1264–1279.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Paheli Bhattacharya, Kaustubh Hiware, Subham Rajgaria, Nilay Pochhi, Kripabandhu Ghosh, and Saptarshi Ghosh. 2019. A comparative study of summarization algorithms applied to legal case judgments. In *European conference on information retrieval*, pages 413–428. Springer.

Naman Chhibbar and Jugal Kalita. 2024. Automatic summarization of long documents. In *Proceedings of the 21st International Conference on Natural Language Processing (ICON)*, pages 607–615, AU-KBC Research Centre, Chennai, India. NLP Association of India (NLPAI).

Debtanu Datta, Shubham Soni, Rajdeep Mukherjee, and Saptarshi Ghosh. 2023. MILDSum: A novel benchmark dataset for multilingual summarization of Indian legal case judgments. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5291–5302, Singapore. Association for Computational Linguistics.

G. Divya and 1 others. 2024. A unified extractive-abstractive framework for long document summarization using bert. *PeerJ Computer Science*.

Mohamed Elaraby, Huihui Xu, Morgan Gray, Kevin Ashley, and Diane Litman. 2024. Adding argumentation into human evaluation of long document abstractive summarization: A case study on legal opinions. In *Proceedings of the Fourth Workshop on Human Evaluation of NLP Systems (HumEval) @ LREC-COLING 2024*, pages 28–35, Torino, Italia. ELRA and ICCL.

Ben Hachey and Claire Grover. 2004. Sentence classification for legal text summarisation. In *Proceedings of the 10th International Conference on Artificial Intelligence and Law*, pages 31–40.

Abhinav Joshi, Shounak Paul, Akshat Sharma, Pawan Goyal, Saptarshi Ghosh, and Ashutosh Modi. 2024. IL-TUR: Benchmark for Indian legal text understanding and reasoning. pages 11460–11499.

Jia Jin Koay, Alexander Roustai, Xiaojin Dai, and Fei Liu. 2021. A sliding-window approach to automatic creation of meeting minutes. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 68–75, Online. Association for Computational Linguistics.

Wojciech Kryściński, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9345.

S. Kumar and 1 others. 2024. A survey on chunking strategies for large language models. *arXiv preprint*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Ziheng Lin, Hwee Tou Ng, and Min-Yen Kan. 2011. Automatically evaluating text coherence using discourse relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 997–1006, Portland, Oregon, USA. Association for Computational Linguistics.

Z. Luo and 1 others. 2024. A survey on factual consistency in the era of large language models. *arXiv preprint arXiv:2402.13758*.

D. Marino and 1 others. 2023. Automatic rhetorical role classification for legal documents using legal-bert. *CEUR Workshop Proceedings*.

L. Miculicich and A. Han. 2023. Document summarization with text segmentation. *arXiv preprint arXiv:2301.08817*.

Gianluca Moro, Luca Ragazzi, Lorenzo Valgimigli, Giacomo Frisoni, Claudio Sartori, and Gustavo Marfia. 2023. Efficient memory-enhanced transformer for long-document summarization in low-resource regimes. *Sensors*, 23(7).

A. Nigam and 1 others. 2025. Legalseg: A large-scale dataset for rhetorical role classification in indian legal judgments. In *Proceedings of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics*.

Pinecone. 2025. Chunking strategies for rag. *Pinecone Learning Center*.

Saisab Sadhu, Biswajit Patra, and Tannay Basu. 2025. Structured adversarial synthesis: A multi-agent framework for generating persuasive financial analysis from earnings call transcripts. In *Proceedings of The 10th Workshop on Financial Technology and Natural Language Processing*, pages 283–291, Suzhou, China. Association for Computational Linguistics.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Saloni Sharma, Surabhi Srivastava, Pradeepika Verma, Anshul Verma, and Sachchida Nand Chaurasia. 2023. A comprehensive analysis of indian legal documents summarization techniques. *SN Computer Science*, 4(6):614.

Abhay Shukla, Paheli Bhattacharya, Soham Poddar, Rajdeep Mukherjee, Kripabandhu Ghosh, Pawan Goyal, and Saptarshi Ghosh. 2022. Legal case document summarization: Extractive and abstractive methods and their evaluation. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1048–1064, Online only. Association for Computational Linguistics.

Josef Steinberger, Massimo Poesio, Mijail A. Kabadjov, and Karel Jeek. 2007. Two uses of anaphora resolution in summarization. *Inf. Process. Manage.*, 43(6):1663–1680.

Y. Sun and 1 others. 2024. Hierarchical abstractive summarization with multi-objective reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Z. Wang and 1 others. 2024. A study on hierarchical information extraction for long text summarization. *Nature*.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2020. PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Wei Zhao, Michael Strube, and Steffen Eger. 2023. DiscoScore: Evaluating text generation with BERT and discourse coherence. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3865–3883, Dubrovnik, Croatia. Association for Computational Linguistics.

Reproducibility

The code, pre-processing scripts, and instructions to reproduce all experiments reported in this paper will be made publicly available at `https://github.com/sonowalh/sac-legal-summ`.

Appendix

A Granular Performance Analysis

To provide quantitative evidence for the "topical drift" phenomenon discussed in the main paper, we conducted a granular, per-section ROUGE analysis. This analysis measures how well each generated summary captures the content of the distinct rhetorical sections of the gold-standard reference summary.

A.1 Methodology

We first manually segmented the 50 reference summaries used for our budget analysis (see Section 3.3.2) into their constituent rhetorical parts: Reference-Facts, Reference-Arguments & Analysis, and Reference-Conclusion. Then, for each of our generated summaries (NSC, SAC-H+, SAC-LLM), we calculated its ROUGE-1 F1-score against each of these three reference segments separately. A high score against Reference-Facts, for example, indicates that the generated summary heavily overlaps with the factual portion of the gold standard.

A.2 Results

The results, presented in Table 5, provide strong numerical evidence for our claims. The NSC summary exhibits a highly skewed performance, achieving a very high ROUGE-1 score of 35.1 against the Reference-Facts but a near-zero score of

2.4 against the Reference-Conclusion. This quantitatively demonstrates topical drift: NSC over-represents the initial facts of the document and almost completely fails to capture the final, critical ruling.

In contrast, both SAC methods show a significantly more balanced performance distribution. They achieve respectable scores across all three sections, with a particularly strong improvement in capturing the Conclusion. This confirms that our structure-aware approach successfully mitigates topical drift and produces a more holistically representative summary.

Method	Ref-Facts	Ref-Arg&An	Ref-Conc
NSC	**35.1**	21.5	2.4
SAC-H+	28.7	25.1	18.9
SAC-LLM	29.2	**26.3**	**19.5**

Table 5: Per-section ROUGE-1 F1 scores comparing full system summaries against individual reference sections. SAC methods produce more balanced coverage than NSC.

B Implementation Details

Heuristic Triggers. The SAC-H method relies on a curated list of regular expression patterns. Table 6 provides a more comprehensive, though not exhaustive, subset of these triggers.

<table>
<tr><th>Section</th><th>Example Trigger Phrases</th></tr>
<tr><td>Arguments & Analysis</td><td>‘learned counsel for the (petitioner|appellant|respondent)’
‘it was contended (by|that)’
‘per contra’
‘the short question which arises’
‘the issue for consideration is’
‘the submission of the learned counsel’
‘it is urged that’</td></tr>
<tr><td>Conclusion</td><td>‘the appeal is (accordingly|partly)? (allowed|dismissed)’
‘the petition is disposed of’
‘for the (above|reasons|aforesaid)’
‘in the result’
‘we are of the considered view’
‘in view of the above discussion’
‘we, therefore, hold that’</td></tr>
</table>

Table 6: A representative subset of high-precision trigger phrases used for rhetorical segmentation in the SAC-H and SAC-H+ models.

SAC-LLM Fallback. The fallback to SAC-H for the rare (<2%) cases where the SAC-LLM method failed to return valid JSON was a pragmatic choice to ensure a fully automated and robust pipeline, preventing the need for manual intervention and maintaining the integrity of the batch evaluation.

From Scratch to Fine-Tuned: A Comparative Study of Transformer Training Strategies for Legal Machine Translation

Amit Barman, Atanu Mandal, Sudip Kumar Naskar
Jadavpur University, Kolkata, INDIA,
Correspondence: amitbarman811@gmail.com

Abstract

In multilingual nations like India, access to legal information is often hindered by language barriers, as much of the legal and judicial documentation remains in English. Legal Machine Translation (L-MT) offers a scalable solution to this challenge by enabling accurate and accessible translations of legal documents. This paper presents our work for the JUST-NLP 2025 Legal MT shared task, focusing on English–Hindi translation using Transformer-based approaches. We experiment with 2 complementary strategies, fine-tuning a pre-trained OPUS-MT model for domain-specific adaptation and training a Transformer model from scratch using the provided legal corpus. Performance is evaluated using standard MT metrics, including SacreBLEU, chrF++, TER, ROUGE, BERTScore, METEOR, and COMET. Our fine-tuned OPUS-MT model achieves a SacreBLEU score of 46.03, significantly outperforming both baseline and from-scratch models. The results highlight the effectiveness of domain adaptation in enhancing translation quality and demonstrate the potential of L-MT systems to improve access to justice and legal transparency in multilingual contexts.

1 Introduction

Since India's independence in 1947, language has remained one of the defining features and challenges of its democracy. The Constitution recognizes 22 scheduled languages, but much of the country's legal, administrative, and judicial work continues to be conducted in English. This linguistic imbalance often leaves citizens dependent on translations to understand laws, judgments, or government notifications that affect their rights. There have been documented instances where individuals have misunderstood court proceedings or official orders simply because they were not available in their native language, an obstacle that runs counter to the ideal of "equal access to justice".

In a multilingual democracy, ensuring that legal information is accessible to all citizens is not only a linguistic challenge but also a civic necessity. Legal texts are particularly complex, they demand precision, consistency, and adherence to jurisdiction-specific terminology. Even small translation errors can lead to misinterpretations, contractual disputes, or procedural delays. As legal materials increasingly move to online platforms, the need for accurate, scalable translation tools has become even more urgent.

Advances in Neural Machine Translation (NMT) have transformed the field of translation, enabling systems to model intricate linguistic relationships and long-range dependencies through attention mechanisms (Vaswani et al., 2017). The rise of Large Language Models (LLMs) trained on vast multilingual data has further improved translation fluency and generalization. Yet, these models often struggle in highly specialized domains like law, where vocabulary, syntax, and semantics diverge significantly from general text. Domain-specific adaptation remains essential for achieving accurate and trustworthy translations.

This paper focuses on developing Legal Machine Translation (L-MT) systems that bridge the linguistic divide in the Indian legal context. As part of the JUST-NLP 2025 Legal MT shared task[1], we investigate how Transformer-based models can be adapted for English–Hindi legal translation. We explore two strategies, training a Transformer model from scratch and fine-tuning the OPUS-MT model, to assess how domain-focused training influences translation quality.

Through this work, we aim to advance the development of reliable and inclusive Legal MT systems that make legal information accessible across languages, supporting transparency, participation, and justice in multilingual societies.

[1] https://exploration-lab.github.io/JUST-NLP/

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 179–185

The key contributions of this paper are summarized as follows:

- We trained and evaluated a Transformer model from scratch on legal-domain data.
- We fine-tuned the Helsinki Opus MT for legal-domain adaptation.
- We analyzed translation robustness and domain adaptability across evaluation datasets.

2 Related Works

Machine Translation (MT) has long been one of the most prominent applications of Natural Language Processing (NLP). Early MT systems were primarily built upon sequence-to-sequence architectures using encoder–decoder frameworks. However, due to their sequential nature and reliance on recurrent neural networks, these models often struggled to capture long-range contextual dependencies effectively.

The introduction of the self-attention mechanism revolutionized MT by enabling models to capture global dependencies among tokens more efficiently. Transformer-based architectures have since become the foundation of modern MT systems, demonstrating exceptional generalization across languages and domains through large-scale multilingual pretraining. This paradigm shift has significantly improved translation fluency, adequacy, and semantic consistency.

Recent advancements in LLMs have further enhanced multilingual translation capabilities through zero-shot and few-shot learning. These pretrained multilingual models can produce reasonable translations even without explicit task-specific fine-tuning. However, their performance tends to degrade substantially for low-resource language pairs, where limited data hampers generalization. To address this, research has increasingly focused on fine-tuning and transfer learning strategies that enable domain and language adaptation. Techniques such as multilingual continued pretraining, cross-lingual embeddings, and parameter-efficient fine-tuning (e.g., adapters like LoRA (Hu et al., 2021) and QLoRA (Dettmers et al., 2023)) have proven effective in improving translation quality for low-resource scenarios. These methods balance computational efficiency with adaptability, allowing pretrained multilingual models to specialize in specific linguistic domains such as legal, medical, or conversational text.

In the Indian context, legal translation has emerged as a crucial area of research due to the nation's linguistic diversity and the absence of a single national language. The growing need to make legal documents accessible across India's many official languages highlights the importance of domain-specific MT systems. However, Indian languages often lack large, high-quality parallel corpora, posing challenges for training robust legal MT models (Joshi et al., 2024).

Over the past decade, several multilingual parallel corpora have been developed for Indian languages. Notable examples include Samanantar (Ramesh et al., 2022), corpus for 11 Indian languages and the corpus by Siripragada et al. (2020), which covers 10 Indian languages. Broader evaluation was also enabled by the FLORES-200 benchmark (Team et al., 2022). Other valuable resources include IndoWordNet (Kunchukuttan, 2020), PMIndia (Haddow and Kirefu, 2020), and datasets such as IITB English-Hindi (Kunchukuttan et al., 2018), BUET English-Bangla (Hasan et al., 2020), English-Tamil (Ramasamy et al., 2012), English-Odia (Parida et al., 2020), and the Mizo-English corpus (Haulai and Hussain, 2023). However, these datasets generally pertain to general-domain translation and are not tailored to the legal domain.

In contrast, the legal domain has seen relatively limited multilingual MT resources. International initiatives such as the Europarl corpus (Koehn, 2005), EUR-Lex (Baisa et al., 2016), and the UN Parallel Corpus (Ziemski et al., 2016), the Bilingwis Swiss Law Text collection (Höfler and Sugisaki, 2014) have provided valuable multilingual datasets for legal proceedings in European languages. However, these resources are largely tailored to European legal systems, linguistic structures, and translation conventions, which differ substantially from the Indian legal and linguistic context. Consequently, such corpora cannot be directly leveraged for Indian-language MT tasks, where distinct terminologies, legal frameworks, and multilingual diversity necessitate domain-specific datasets and adaptation strategies.

Within India, only a handful of initiatives have attempted to build legal-domain corpora. The Hindi–Telugu legal dataset from LTRC (Mujadia and Sharma, 2022) and the Anuvaad corpus[2] repre-

[2]https://github.com/project-anuvaad/anuvaad-parallel-corpus

sents early efforts, however, they lack expert validation. The recently introduced MILPaC corpus (Mahapatra et al., 2025) marks a significant advancement, offering a well-curated, expert-validated, and multilingual benchmark for legal MT. Additionally, the WMT25 Legal Domain Test Suite (Singh et al., 2025) provides a robust evaluation framework for assessing MT capabilities in English–Hindi legal translation. Together, these initiatives represent an emerging but still underdeveloped ecosystem for legal-domain MT in Indian languages.

3 Dataset Description

Table 1 summarizes the dataset used in this study. Provided by the task organizers, it consists of English–Hindi parallel sentence pairs from the legal domain. Only the training pairs were initially released, while validation and test references were withheld. Participants generated translations for these sets during the evaluation and final phases, with the reference translations revealed after the leaderboard announcement.

Table 1: Dataset statistics for Legal Machine Translation (L-MT) Shared Task

Language Pair	Train	Validation	Test
English-Hindi	50,000	5,000	5,000

The dataset contains 60,000 English–Hindi parallel sentences from the legal domain, divided into 50,000 for training and 5,000 each for validation and testing. The wide variation in sentence length suggests diverse syntactic structures typical of legal text. While the dataset is well-balanced and cleanly split, the absence of metadata on text type or source (e.g., statutes, judgments, or general documents) limits fine-grained domain analysis.

4 Experiments

Transformer-based architectures have become the foundation of modern NMT due to their ability to model complex contextual relationships through self-attention. They outperform traditional sequence-to-sequence models, particularly in tasks requiring structural precision and contextual awareness, which are vital in legal translation. However, large Transformer models are computationally expensive. For this study, we employed two complementary training strategies suitable for constrained resources:

- Fine-tuning a pre-trained OPUS-MT model to adapt general translation knowledge to the legal domain, and
- Training a Transformer from scratch to evaluate its capability to learn domain-specific patterns directly from legal text.

Our experimental setup is available in the following Link[3].

4.1 Opus Fine-Tune

We fine-tuned the Helsinki Opus-MT model[4] using the provided training corpus. Since validation references were initially withheld, evaluation was based on interim submissions. The model was optimized using the AdamW optimizer with a learning rate of 2×10^{-5}, weight decay 0.01, and batch size 32. Both input and target sequences were limited to 128 tokens to ensure computational efficiency without excessive truncation. As previous research (Cho et al., 2014) suggests, excessively long inputs degrade model performance due to weakened attention over long dependencies, hence, this cap provides an effective trade-off between fidelity and efficiency.

4.2 Transformer Training

To evaluate the impact of learning solely from domain-specific data, we trained a compact Transformer model from scratch. The configuration included 4 encoder-decoder layers, 8 attention heads, model dimension of 128, dropout of 0.1, and token length of 256, and a vocabulary size derived from a SentencePiece tokenizer of 32,000. The model was trained with a batch size of 32, using the Adam optimizer. Despite limited data, this model demonstrated strong convergence, underscoring the ability of smaller Transformers to effectively learn domain-specific translation patterns when carefully optimized.

4.3 Evaluation Setup

Model outputs were assessed using multiple metrics capturing lexical, syntactic, and semantic correspondence: SacreBLEU (Papineni et al., 2002; Post, 2018), chrF++ (Popović, 2015), TER (Snover et al., 2006), ROUGE (Lin, 2004), BERTScore (Zhang et al., 2019), METEOR (Lavie and Agarwal, 2007; Banerjee and Lavie, 2005),

[3]https://github.com/atanumandal0491/Legal-Translation
[4]Helsinki Opus-MT

Table 2: Final leaderboard results for the JUST-NLP 2025 Shared Task on Legal Machine Translation (English-Hindi). The best scores for each metric are highlighted. Our system (JUNLP) achieved Rank 4 with competitive performance across lexical and semantic metrics.

Rank	Team Name	Country	BLEU ↑	chrF++ ↑	TER ↓	BERTScore (F1) ↑	METEOR ↑	COMET ↑	AutoRank ↑
1	Team-SVNIT	India	**51.61**	73.29	**37.09**	92.61	**75.80**	**76.36**	**61.62**
2	FourCorners	Thailand	50.19	**73.67**	42.32	**92.70**	69.54	75.74	60.31
3	goodmen	India	48.56	73.07	41.63	92.38	67.15	75.16	59.39
4	JUNLP	India	46.03[6]	70.59[4]	42.08[3]	91.19[4]	71.84[3]	73.72[4]	58.90
5	JUST-MEI	India	46.67	70.03	44.63	90.86	72.86	72.12	58.79
6	Lawgorithms	India	46.27	68.32	43.06	91.03	71.80	72.14	58.26
7	Tokenizers	India	34.08	56.75	55.25	87.39	61.78	65.20	50.87

Table 3: Comparison of translation performance across different models on the English-Hindi legal dataset.

Model	Fine-Tuned	BLEU ↑	chrF++ ↑	TER ↓	ROUGE-1 ↑	ROUGE-2 ↑	ROUGE-L ↑	BERTScore (F1) ↑	METEOR ↑	COMET ↑
OPUS-MT (fine-tuned)	✓	**46.03**	**70.59**	**42.08**	**72.42**	**52.63**	**69.05**	**91.19**	**71.84**	**73.72**
OPUS-MT (baseline)	✗	9.39	27.66	83.40	36.30	13.38	32.93	76.91	30.25	50.80
Transformer (trained from scratch)	✗	37.77	60.88	59.72	35.98	13.62	35.69	88.37	65.58	64.29
NLLB (3.3B distilled)	✗	23.72	47.50	63.29	49.00	26.31	45.78	85.14	45.32	67.25
IndicTrans2	✗	10.87	42.36	81.25	37.89	11.07	37.10	81.21	41.78	60.38

and COMET (Rei et al., 2020). These complementary measures ensure robust evaluation across the dimensions of precision, recall, fluency, and semantic alignment.

5 Results and Analysis

Table 2 presents the final leaderboard results from the JUST-NLP 2025 Shared Task on Legal Machine Translation, comparing the performance of participating systems across a range of lexical, semantic, and edit-based evaluation metrics. Our system, JUNLP, achieved an overall Rank 4, with a SacreBLEU score of 46.03, chrF++ of 70.59, and TER of 42.08, demonstrating strong translation accuracy, requiring relatively low post-editing effort. The model also performed competitively in semantic evaluation, achieving a BERTScore (F1) of 91.19, METEOR of 71.84, and COMET of 73.72, indicating high alignment with human reference translations. While the best-performing team attained marginally higher results across several metrics, our system performed in the mid-range compared to the other participating systems (cf. Table 3), underscoring the effectiveness of domain-focused fine-tuning for legal translation.

Table 3 summarizes our experimental outcomes, comparing the fine-tuned OPUS-MT models with baseline multilingual models. The baseline OPUS-MT (without fine-tuning) performed poorly, with a SacreBLEU of 9.39 and chrF++ of 27.66, revealing significant deviation from reference translations. BERTScore F1 of 76.91 and a COMET score of 50.8 further indicate weak semantic alignment and limited adaptability of the baseline OPUS-MT to the legal domain.

The fine-tuned OPUS-MT markedly improved translation quality, achieving a SacreBLEU of 46.03, chrF++ of 70.59, and TER of 42.08, demonstrating high lexical accuracy and fluency. The BERTScore (91.19) and COMET (73.72) show strong semantic alignment with human references, while METEOR (71.84) and ROUGE scores confirm consistent n-gram and paraphrase correspondence. These results suggest that fine-tuning effectively transfers linguistic and contextual knowledge from general corpora to specialized legal data without overfitting. This performance reinforces the viability of fine-tuning for domain-specific translation and motivates further exploration of scalable approaches such as parameter-efficient tuning and extension to additional Indian languages.

The Transformer model trained from scratch performed competitively, achieving a SacreBLEU of 37.77 and COMET of 64.29. Despite lacking pre-trained initialization, it captured domain patterns effectively, although the fine-tuned OPUS-MT maintained an edge in fluency and semantic coherence. Multilingual baselines, such as NLLB and

IndicTrans2, performed moderately, underscoring that general-purpose models struggle with domain-specific precision.

Overall, the fine-tuned OPUS-MT model produced fluent, accurate, and contextually faithful translations, confirming its effectiveness for English–Hindi legal MT in real-world settings.

6 Conclusion and Future Work

This work explored domain adaptation strategies for Legal Machine Translation (L-MT) in the English-Hindi context, highlighting how fine-tuning enhances translation quality for specialized text. Among all systems tested, the fine-tuned OPUS-MT model achieved the highest performance, demonstrating superior lexical accuracy and semantic consistency. Training a Transformer model from scratch also yielded promising results, showing that domain-specific supervision alone can produce competitive results under constrained resources.

Future work will extend these experiments to other Indian languages and evaluate parameter-efficient fine-tuning techniques such as LoRA and QLoRA to scale Legal MT further. Ultimately, such systems can play a transformative role in democratizing access to legal knowledge, ensuring that linguistic diversity does not become a barrier to justice.

Limitations

While the proposed approach demonstrates strong empirical performance, several limitations constrain the generalizability and scope of the current study:

- **Restricted training corpus:** The model was trained exclusively on the dataset released by the shared task organizers, without augmentation from external legal or general-domain corpora. Consequently, the system's exposure to broader linguistic variability and complex domain phenomena remains limited.

- **Lack of comprehensive validation data:** Complete source-target validation pairs were unavailable during training, which hindered reliable monitoring of model behavior (e.g., overfitting or underfitting) and constrained opportunities for principled hyperparameter optimization.

- **Sequence length constraints:** Input-output sequences were truncated to a maximum of 128 tokens due to computational limitations. Although suitable for most sentence-level examples, this restriction may adversely affect the processing of lengthy statutory clauses, compound sentences, or cross-referential structures.

- **Absence of human evaluation:** The assessment relies primarily on automated metrics (SacreBLEU, chrF++, BERTScore, COMET), and does not incorporate expert human judgement, limiting deeper qualitative insights into adequacy, legal constancy, and pragmatic interpretability.

- **Resource constraints:** Due to time and computational constraints, broader experimental exploration, such as parameter-efficient tuning, multilingual transfer, or alternative architectures-was not undertaken.

Acknowledgments

This research was funded by the 'VIDYAAPATI: Bidirectional Machine Translation Involving Bengali, Konkani, Maithili, Marathi, and Hindi' under the Project titled 'National Language Translation Mission (NLTM): BHASHINI'.

References

Vít Baisa, Jan Michelfeit, Marek Medveď, and Miloš Jakubíček. 2016. European Union language resources in Sketch Engine. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2799–2803, Portorož, Slovenia. European Language Resources Association (ELRA).

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. *Preprint*, arXiv:1409.1259.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms.

Barry Haddow and Faheem Kirefu. 2020. Pmindia – a collection of parallel corpora of languages of india. *Preprint*, arXiv:2001.09907.

Tahmid Hasan, Abhik Bhattacharjee, Kazi Samin, Masum Hasan, Madhusudan Basak, M. Sohel Rahman, and Rifat Shahriyar. 2020. Not low-resource anymore: Aligner ensembling, batch filtering, and new datasets for Bengali-English machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2612–2623, Online. Association for Computational Linguistics.

Thangkhanhau Haulai and Jamal Hussain. 2023. Construction of mizo: English parallel corpus for machine translation. 22(8).

Stefan Höfler and Kyoko Sugisaki. 2014. Constructing and exploiting an automatically annotated resource of legislative texts. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 175–180, Reykjavik, Iceland. European Language Resources Association (ELRA).

J. Edward Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *ArXiv*, abs/2106.09685.

Abhinav Joshi, Shounak Paul, Akshat Sharma, Pawan Goyal, Saptarshi Ghosh, and Ashutosh Modi. 2024. IL-TUR: Benchmark for Indian legal text understanding and reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11460–11499, Bangkok, Thailand. Association for Computational Linguistics.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of Machine Translation Summit X: Papers*, pages 79–86, Phuket, Thailand.

Anoop Kunchukuttan. 2020. Indowordnet parallel corpus. `https://github.com/anoopkunchukuttan/indowordnet_parallel`.

Anoop Kunchukuttan, Pratik Mehta, and Pushpak Bhattacharyya. 2018. The IIT Bombay English-Hindi parallel corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Alon Lavie and Abhaya Agarwal. 2007. Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. pages 228–231.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Sayan Mahapatra, Debtanu Datta, Shubham Soni, Adrijit Goswami, and Saptarshi Ghosh. 2025. Milpac: A novel benchmark for evaluating translation of legal text to indian languages. 24(8).

Vandan Mujadia and Dipti Sharma. 2022. The LTRC Hindi-Telugu parallel corpus. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3417–3424, Marseille, France. European Language Resources Association.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Shantipriya Parida, Satya Ranjan Dash, Ondřej Bojar, Petr Motlicek, Priyanka Pattnaik, and Debasish Kumar Mallick. 2020. OdiEnCorp 2.0: Odia-English parallel corpus for machine translation. In *Proceedings of the WILDRE5– 5th Workshop on Indian Language Data: Resources and Evaluation*, pages 14–19, Marseille, France. European Language Resources Association (ELRA).

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Loganathan Ramasamy, Ondrej Bojar, and Z. Žabokrtský. 2012. Morphological processing for english-tamil statistical machine translation.

Gowtham Ramesh, Sumanth Doddapaneni, Aravinth Bheemaraj, Mayank Jobanputra, Raghavan AK, Ajitesh Sharma, Sujit Sahoo, Harshita Diddee, Mahalakshmi J, Divyanshu Kakwani, Navneet Kumar, Aswin Pradeep, Srihari Nagaraj, Kumar Deepak, Vivek Raghavan, Anoop Kunchukuttan, Pratyush Kumar, and Mitesh Shantadevi Khapra. 2022. Samanantar: The largest publicly available parallel corpora collection for 11 Indic languages. *Transactions of the Association for Computational Linguistics*, 10:145–162.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.

Kshetrimayum Boynao Singh, Deepak Kumar, and Asif Ekbal. 2025. Evaluation of LLM for English to Hindi legal domain machine translation systems. In *Proceedings of the Tenth Conference on Machine Translation*, pages 823–833, Suzhou, China. Association for Computational Linguistics.

Shashank Siripragada, Jerin Philip, Vinay P. Namboodiri, and C V Jawahar. 2020. A multilingual parallel corpora collection effort for Indian languages. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3743–3751, Marseille, France. European Language Resources Association.

Matthew Snover, Bonnie Dorr, Rich Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 223–231, Cambridge, Massachusetts, USA. Association for Machine Translation in the Americas.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, and 20 others. 2022. No language left behind: Scaling human-centered machine translation. *Preprint*, arXiv:2207.04672.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *ArXiv*, abs/1904.09675.

Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The United Nations parallel corpus v1.0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3530–3534, Portorož, Slovenia. European Language Resources Association (ELRA).

Integrating Graph based Algorithm and Transformer Models for Abstractive Summarization

Sayed Ayaan Ahmed Sha*, Sangeetha Sivanesan*,
Anand Kumar Madasamy+, Navya Binu+
*National Institute of Technology, Tiruchirappalli
+National Institute of Technology karnataka, Surathkal
405425002@nitt.edu

Abstract

Summarizing legal documents is a challenging and critical task in the field of Natural Language Processing(NLP). On top of that generating abstractive summaries for legal judgments poses a significant challenge to researchers as there is limitation in the number of input tokens for various language models. In this paper we experimented with two models namely BART base model finetuned on CNN DailyMail dataset along with TextRank and `pegasus_indian_legal`, a finetuned version of legal-pegasus on indian legal judgments for generating abstractive summaries for Indian legal documents as part of the **JUST-NLP 2025 - Shared Task on Legal Summarization**. BART+TextRank outperformed `pegasus_indian_legal` with a score of 18.84.

1 Introduction

Legal texts have special characteristics that set them apart from other document. Compared to other domains, legal documents are typically longer and more thorough. They employ a lot of domain-specific jargon, acronyms, and citations or references, and their language is complicated(Akter et al., 2025). Summarization of legal documents is a fundamental and challenging task in legal practice as cases are complex and lengthy in nature(Shukla et al., 2022). A majority of India's population lacks a strong command of the English language (Datta et al., 2023) and it is difficult for a layman to comprehend the complex structure of a judgment. Thus, it is crucial to summarize legal documents in plain and understandable english. Legal text summarization focuses on extracting and highlighting the essential points of a legal document in a brief and clear manner, enabling quick decision-making. Creating technologies that can handle documents from specific nations or languages is one of the targets of region-specific legal summarization(Akter et al., 2025). Region-specific legal documents, summarization strategies, and summarization methodologies are the three basic components of legal summarization.

General summarization strategies identified in legal summarization involve extractive, abstractive, and hybrid approaches. The extractive method(Cheng and Lapata, 2016) involves directly copying significant sentences from the source document and combining them to create the output summary. Meanwhile the abstractive strategy(Rush et al., 2015) mimics human understanding by interpreting the source document and producing a summary based on its important concepts. By reconstructing a summary using certain key information taken from the original document, the hybrid technique seeks to combine the advantages of both approaches. However, there are a variety of summarization techniques, including rank-based, graph-based, transformer-based, and others.

2 Related Work

Wide range of solutions have been proposed for the summarization of indian as well as foreign legal documents. Among the extractive family, there are unsupervised summarization approaches such as Reduction(Jing, 2000) and graphical approach LexRank(Erkan and Radev, 2004). SummaRuNNer(Nallapati et al., 2017) and BERTSum(Liu and Lapata, 2019) are two supervised neural summarizers that approach document summarization as a binary classification issue (in-summary vs. out-of-summary). Some abstractive models include (Lewis et al., 2020) and (Zhang et al., 2020). Generating abstractive summaries specifically for indian legal documents, (Shukla et al., 2022) created the IN-Abs corpus containing 7,130 case documents, together with their headnotes/summaries. IL-TUR(Joshi et al., 2024) a benchmark for legal tasks, included the SOTA models for indian legal

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 186–190
December 24, 2025 ©2025 Association for Computational Linguistics

text summarization. One of the early works in legal text summarization include identification of the thematic structure to find the argumentative themes of the judgement(Farzindar and Lapalme, 2004). The relevant sentences were extracted for each theme and presented as a table-style summary. (Verma et al., 2022) suggested a two-step method in which sentences were first clustered by similarity using a partitional technique; key sentences from each cluster were then chosen based on text feature scores, and similarity was measured using a linear combination of normalised Google distance and word mover's distance.

3 Dataset

The **InLSum**(Indian Legal Summarization) dataset was provided by the organizers of the workshop. It had 3 splits - train (1200 datapoints), validation (200 datapoints), and test (400 datapoints). Each entry correspond to one court judgment with a unique ID and with two files for each split (train, validation, test): each split contained the full text judgments and the other contained the gold(reference) summaries. Each summary is an abstractive human-written summary of the respective judgment.

4 Task and Evaluation

The task focused on generating abstractive summaries for the indian court judgments in English. Participants were required to train language models that can read legal judgments and produce concise, coherent, and fluent summaries of approximately 500 words. The models were trained on the training split of the dataset and the predictions for the validation set and train set were submitted for evaluation.

The submitted predictions were evaluated by the organizers using three standard metrics: (i) ROUGE-2, (ii) ROUGE-L (Lin, 2004) and (iii) BLEU(Papineni et al., 2002).

5 Experimentation

For producing summaries for the given dataset, we experimented with two approaches: (i) We use BART (Lewis et al., 2020) base model fine-tuned on DailyMail dataset[1] along with TextRank(Mihalcea and Tarau, 2004) for generating the desired output summaries. Bart leverages a standard seq2seq architecture with a bidirectional encoder(like BERT(Devlin et al., 2019)) and a left-to-right decoder (like GPT(Radford et al., 2018)). BART has proven to produce effective results when finetuned for text generation tasks. TextRank(Mihalcea and Tarau, 2004) is a graphical unsupervised extractive summarization strategy. It leverages embeddings for generating the similarity scores between the sentences and stores them in a matrix. The similarity matrix is then converted into a graph, with sentences acting as vertices and the scores as edges for computing the sentence rank. (ii) `pegasus_indian_legal`[2], a fine-tuned version of legal-pegasus on indian legal judgments. The legal-pegasus is also a further finetuned version of Pegasus(Zhang et al.). We have provided the github link for the codes and experiments performed for this paper[3]

5.1 Training and Validation phase

For the first approach, we first finetuned the BART-base-cnn model on the training data and then used the TextRank algorithm to extract top ranked sentences from the validation set. We had input token constraint of 1024 and assigned generated summary length to maximum of 768 tokens. To generate the embeddings for the judgments, we utilized all-MiniLM-L6-v2, a sentence transformer model. For each judgment, we compute a similarity matrix from the sentence embeddings for it's constituent sentences. The similarity scores were computed using cosine similarity. We select top n sentences for each judgment based on the scores using TextRank and the resultant sentences were fed to the model for generating the abstractive summaries. To choose the optimal n value, we first calculated the average number of sentences in the reference summaries of train set and test set and found it to be approximately 20. We experimented with n values 20 and 30. When n = 30, the results for the validation set were slightly lower when n is 20. The n value we finally opted was 20.

In the second approach, we finetuned the pegasus_indian_legal model on the train dataset provided by the organizers and ran the finetuned model on the validation set to generate the summary. Here also the maximum input token sequence is 1024 tokens and the summary length was intilalized to

[1]`https://huggingface.co/ainize/bart-base-cnn`

[2]`https://huggingface.co/akhilm97/pegasus_indian_legal`

[3]`https://github.com/Ayaan2123/Shared_Task-2025`

maximum of 768 tokens.

During the training phase, the HuggingFace implementation of BART/pegasus_indian_legal applies automatic truncation when the input exceeds the maximum length of the models. Thus, for input sequences longer than 1024 tokens, the models only used the first 1024 tokens.

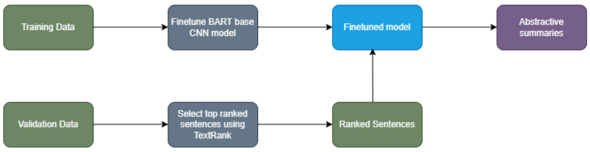


Figure 1: Summary Generation Workflow

5.2 Testing Phase

The task was to generate abstractive summaries of approximately 500 words for the test data provided by the organizers. The respective finetuned models were finally tested using the test set.

6 Evaluation and Analysis

The organizers used three set of standard metrics for evaluating the performance of the summarization models: (i) ROUGE-2 (ii) ROUGE-L (Lin, 2004) and (iii) BLEU(Papineni et al., 2002). Our work finished 9th in the final rankings with an average score of 18.84

BART+TextRank proved to be the best performer among the two approaches we incorporated for the desired results. Table 1 gives the performance of the models on the validation as well as test set. BART along with TextRank performs better on validation data while it drops slightly in the testing phase. The first approach still yields better results than PEGASUS_INDIAN_LEGAL. Appendix A provides sample summaries generated for a particular judgment using both the approaches.

Algorithm	R-2	R-L	BLEU	Average
Performance during validation phase				
BART + TEXTRANK	21.37	22.42	15.64	19.81
PEGASUS_INDIAN_LEGAL	14.39	23.08	9.19	15.55
Performance during testing phase				
BART + TEXTRANK	20.37	22.49	13.67	18.84
PEGASUS_INDIAN_LEGAL	15.64	23.27	11.75	16.89

Table 1: Performance comparison of summarization methods.

7 Conclusion

Summarizing legal documents is a critical task, facing significant challenges due to the volume and complexity of legal documents. In this paper we tested two models for generating the abstractive summaries for Indian legal judgments and analyzed their performance. There is a lot of room for future development and progress in the area of legal document summarization.

Limitations

Our work has several limitations that should be considered during the results analysis. Both BART and PEGASUS models are constrained by a maximum input length of 1024 tokens, while the legal judgments in the dataset are significantly longer. In the first approach (BART+TextRank), we partially mitigated this problem during inference by using an extractive then abstractive pipeline. However, in Approach 2 (PEGASUS_INDIAN_LEGAL), the model relied on the truncation behavior of the tokenizer, which limits the model's ability to reason over the entire judgment. This disparity in long-context handling likely contributed to the performance differences between the two approaches.

Acknowledgments

We extend our gratitude to the organizers of the **JUST-NLP 2025 - Shared Task on Legal Summarization** for providing the data and evaluation resources. We also thank our team members for their assistance and valuable contributions.

References

Mousumi Akter, Erion Çano, Erik Weber, Dennis Dobler, and Ivan Habernal. 2025. A comprehensive survey on legal summarization: Challenges and future directions. *arXiv preprint arXiv:2501.17830*.

Jianpeng Cheng and Mirella Lapata. 2016. Neural summarization by extracting sentences and words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 484–494, Berlin, Germany. Association for Computational Linguistics.

Debtanu Datta, Shubham Soni, Rajdeep Mukherjee, and Saptarshi Ghosh. 2023. MILDSum: A novel benchmark dataset for multilingual summarization of Indian legal case judgments. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5291–5302, Singapore. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the*

North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers), pages 4171–4186.

Günes Erkan and Dragomir R Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of artificial intelligence research*, 22:457–479.

Atefeh Farzindar and Guy Lapalme. 2004. Letsum, an automatic legal text summarizing. *Legal knowledge and information systems: JURIX*, page 11.

Hongyan Jing. 2000. Sentence reduction for automatic text summarization. In *Sixth Applied Natural Language Processing Conference*, pages 310–315, Seattle, Washington, USA. Association for Computational Linguistics.

Abhinav Joshi, Shounak Paul, Akshat Sharma, Pawan Goyal, Saptarshi Ghosh, and Ashutosh Modi. 2024. IL-TUR: Benchmark for Indian legal text understanding and reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11460–11499, Bangkok, Thailand. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3730–3740, Hong Kong, China. Association for Computational Linguistics.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411, Barcelona, Spain. Association for Computational Linguistics.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, and 1 others. 2018. Improving language understanding by generative pre-training.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

Abhay Shukla, Paheli Bhattacharya, Soham Poddar, Rajdeep Mukherjee, Kripabandhu Ghosh, Pawan Goyal, and Saptarshi Ghosh. 2022. Legal case document summarization: Extractive and abstractive methods and their evaluation. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1048–1064, Online only. Association for Computational Linguistics.

Pradeepika Verma, Anshul Verma, and Sukomal Pal. 2022. An approach for extractive text summarization using fuzzy evolutionary and clustering algorithms. *Applied Soft Computing*, 120:108670.

J Zhang, Y Zhao, M Saleh, and PJ Liu. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. arxiv 2019. *arXiv preprint arXiv:1912.08777*.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 11328–11339. PMLR.

Appendix

A Comparison of generated summaries

BART+TextRank	PEGASUS_INDIAN_LEGAL
A proposed construction project at NOIDA which did not take off from the drawing board has given rise to proceedings under Article 32 of the Constitution by twenty-five purchasers of commercial premises. The Court made the following observations:The writ petition requires the Court to step into the construction project and ensure that it is duly completed. This would be beyond the remit and competence of the Court under the Constitution."It would be inappropriate for this Court to assume the jurisdiction to supervise the due completion of a construction project especially in facts such as those presented in the present case. This will inevitably draw the Court into the day-to-day supervision of the project, including financing, permissions, and execution—something which lies beyond the ken of judicial review and the competence of a court," the Court said. "The Court must confine itself to its core competencies which consist in the adjudication of disputes amenable to the application of legal standards," the Court added.It added that the writ petition is disposed of, subject to the aforesaid liberty.It was also suggested that the decision of a public authority entrusted with a public duty is amenable to judicial review. But it is quite another hypothesis to postulate that the decision-making authority should be taken over by the Court. The latter is impermissible." The petitioners have sought, inter alia, a writ, order or direction to the respondents to protect the interests and investments of customers/buyers in the larger public interest.A petition application, if any, stands disposed of, the Court noted.In the case, the petitioners had sought to pursue remedies available at law, including the Consumer Protection Act 1986, the Real Estate (Regulation and Development) Act 2016, and the Insolvency and Bankruptcy Code 2016.On the other hand, the Court noted that several provisions of law confer statutory rights on buyers of real estate and invest them with remedies enforceable at law.There was no mention of the clause in the Constitution, which provides for a statutory regime to protect purchasers and created fora which are entrusted with decision-making power.However, it was contended that a decision of a public authority entrusted with public duty was amenable to judicial review, but it is impermissible to take over the decision-making authority. The petitioners were represented by Shiv Kumar Suri and Nikita Thapar.While the Court was hearing the matter, it said that it was not appropriate to assume jurisdiction of the Court under Article 32 of the Constitution."We leave it open to the petitioners to pursue the remedies available in law," the Court said.The writ petition was disposed of and the petition application stood disposed of. The Court added that it would not be appropriate for the Court to take jurisdiction into day-to-day supervision of such a project.The Court added that a writ petition filed by the petitioner was disposed of subject to liberty.Petitioners also submitted that the petition filed against the petitioner stood dismissed and the case was adjourned until the matter was heard by the Court on Thursday."A decision of the petitioner would not be disposed of if the petition was filed before the Court. It was submitted by the counsel of Vinishma Kaul."	The Delhi High Court recently dismissed a petition filed by a group of real estate investors seeking a court-monitored probe into a proposed construction project at NOIDA.A bench of Chief Justice of India (CJI) Dipak Misra and Justice Sanjay Kishan Kaul said that managing a construction project is not within the jurisdiction of the Court under Article 32 of the Constitution. "It would be inappropriate for this Court to assume the jurisdiction to supervise the due completion of a construction project especially in facts such as those presented in the present case.This will inevitably draw the court into the day-to-day supervision of the project, including financing, permissions and execution – something which lies beyond the ken of judicial review and the competence of the court.The court must confine itself to its core competencies which consist in the adjudication of disputes amenable to the application of legal standards.We, therefore, leave it open to the petitioners to pursue the remedies available in law," the Court said.The Court was hearing a petition filed by a group of real estate investors seeking a court-monitored probe into a proposed construction project at NOIDA. The petitioners claimed that the project did not take off from the drawing board and had given rise to proceedings under Article 32 of the Constitution by twenty five purchasers of commercial premises.They sought a writ, order or direction to the respondents to protect the interests and investments of customers/buyers in the larger public interest; (ii) a writ, order or direction for the revival of the project failing which the amounts invested by the petitioners be returned with interest at the rate of 18 % per annum; and (iii) a court-monitored probe.The Court was of the view that the exercise of jurisdiction under Article 32 of the Constitution would not be warranted in the facts of the present case."The writ petition requires the Court to step into the construction project and to ensure that it is duly completed. This would be beyond the remit and competence of the Court under Article 32. Managing a construction project is not within the jurisdiction of the court. Several provisions of law confer statutory rights on purchasers of real estate and invest them with enforceable remedies at law.Several provisions of law confer a statutory regime to protect the rights of purchasers of real estate and created fora which are entrusted with decision-making authority," the Court said."A decision of a public authority which is entrusted with a public duty is amenable to judicial review.But it is quite another hypothesis to postulate that the decision-making authority should be taken over by the court.The latter is impermissible.The latter is impermissible," the Court said.The Court was of the view that managing a construction project is not within the remit and competence of the Court under Article 32 of the Constitution."The court must confine itself to its core competencies which consist in the adjudication of disputes amenable to the application of legal standards.We, therefore, leave it open to the petitioners to pursue the remedies available in law," the Court said.The Court was hearing a petition filed by Shikhil Suri, Shiv Kumar Suri, Madhu Suri, Shilpa Saini, Nikita Thapar, Vinishma Kaul and Vinishma Kaul.

Table 2: Summary comparison of the two methods.

Hierarchical Long-Document Summarization using LED for Legal Judgments

Reshma Sheik[1] **Noah John Puthayathu**[1] **Fathima Firose A**[2] **Jonathan Paul**[1]
[1]TKM College of Engineering, Kollam [2]Thiagarajar College of Engineering, Madurai
rezmasheik@gmail.com,230979@tkmce.ac.in,
fathimafiroseofficial@gmail.com,230685@tkmce.ac.in

Abstract

This paper describes our system for the L-SUMM shared task on legal document summarization. Our approach is built on the Longformer Encoder-Decoder (LED) model, which we augment with a multi-level summarization strategy tailored for legal documents that are substantially longer than typical transformer input limits. The system achieved competitive performance on the legal judgment summarization task through optimized training strategies, including gradient accumulation, Adafactor optimization, and hyperparameter tuning. Our findings indicate that combining hierarchical processing with strategically assigned global attention enables more reliable summarization of lengthy legal texts.

1 Introduction

Summarizing legal documents presents unique challenges due to the length, complexity, and specialised terminology of legal texts. Judicial rulings can span many thousands of words and include intricate argumentation, citations, and procedural information that should be retained in any condensed representation. Recent breakthroughs in transformer-based models have made significant progress in handling long documents more effectively (Lewis et al., 2019; Zhang et al., 2020). The Longformer Encoder-Decoder (LED) architecture extends transformers to handle sequences up to 16,384 tokens through efficient attention patterns (Beltagy et al., 2020). However, many legal judgments exceed even this extended context window.

We present a hierarchical summarization system that combines LED's long-context capabilities with a chunk-and-aggregate approach for extremely long documents. We evaluate our system on the L-SUMM shared task using the InLSum dataset. Our contributions include:

- A robust preprocessing pipeline handling diverse legal text formatting
- Hierarchical summarization for documents exceeding model capacity
- Extensive hyperparameter optimization for legal domain adaptation

2 Related Work

2.1 Efficient Attention Mechanisms for Long Sequences

Traditional transformer models are limited to 512 or 1024 tokens because their self-attention mechanism scales quadratically with sequence length (Vaswani et al., 2017). Several approaches have been proposed to extend this capacity, including sparse attention patterns (Zaheer et al., 2020), retrieval-augmented methods (Lewis et al., 2020), and hierarchical architectures (Cohan et al., 2018).

The Longformer model employs a mixed attention pattern, combining local window-based attention with global tokens to efficiently model long sequences. These architectures have proven to be effective for long-document understanding and summarization tasks.

2.2 Legal Document Processing

Legal NLP has seen increasing interest, with shared tasks focusing on case outcome prediction, statute retrieval, and summarization (Kano et al., 2018; Chalkidis et al., 2020). Legal texts require domain-specific handling due to their length, formal language, and citation structure.

Previous surveys highlight the challenges of summarizing lengthy legal judgments, emphasizing the need for specialized models capable of handling complex reasoning and domain-specific terminology (Kanapala et al., 2019).

2.3 Indian Legal Document Summarization

Automatic summarization and structural analysis of Indian legal judgments have attracted attention

Proceedings of the 1st Workshop on NLP for Empowering Justice (JUST-NLP 2025), pages 191–195
December 24, 2025

in recent years, as the large volume and complexity of judgments pose challenges to accessibility and comprehension. Previous work such as Bhattacharya et al. (2019) studied rhetorical role classification of sentences in judgments from the Indian Supreme Court, allowing the segmentation of legal documents into semantically meaningful segments such as facts, issues, reasoning, and rulings, which is a useful preprocessing step for summarization and retrieval. Bhattacharya et al. (2021) proposed an unsupervised summarization method (DELSumm) for Indian case documents, showing promising ROUGE scores without the need for large annotated datasets. Building on dataset creation efforts, Parikh et al. (2021) released a corpus of over 10,000 Indian judgments paired with handwritten summaries and developed weakly supervised summarization baselines. Our work extends these efforts by combining hierarchical processing with long-context transformer architectures to handle extremely long judgments, a challenge common in the Indian legal domain.

3 System Description

Our system employs a hybrid approach that adapts to document length. Figure 1 provides an overview of the complete pipeline, showing both the direct summarization path for shorter documents and the hierarchical approach for longer ones.

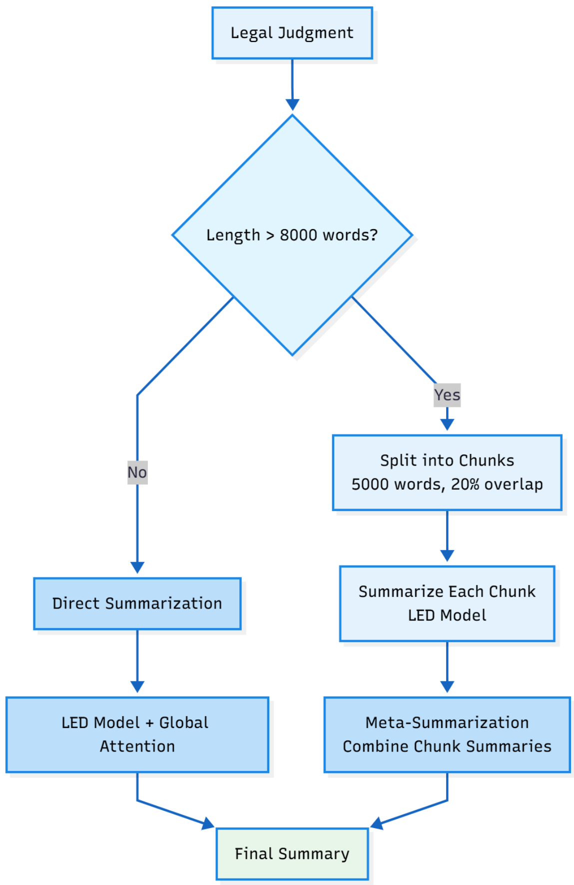


Figure 1: System pipeline showing adaptive processing based on document length.

3.1 Model Architecture

We employ the LED-large-16384 model[1] as our base architecture. LED extends the Longformer's efficient attention mechanism to seq2seq tasks, combining:

- **Local attention**: Sliding window attention with window size 512
- **Global attention**: Selected tokens that attend to all positions
- **Encoder-decoder structure**: Enabling abstractive generation

Our implementation configures global attention on the first 64 tokens and then periodically every 384 tokens throughout the document. This pattern was selected after examining the structure of the legal judgments in the dataset, where important elements such as party names, issue statements, statutory references, and section headings recur at regular intervals. Assigning global attention at these periodic positions ensures that structurally significant segments are consistently captured.

3.2 Data Preprocessing

Legal judgments contain various formatting artifacts that require careful preprocessing:

1. **Page number removal**: Patterns like [Page No. X] are stripped
2. **Whitespace normalization**: Excessive newlines and spaces are compressed
3. **Column name standardization**: All metadata fields are lowercased for robust schema handling

The preprocessing ensures clean input while preserving the semantic content and structure of legal arguments.

[1]https://huggingface.co/allenai/led-large-16384

3.3 Hierarchical Summarization

For documents exceeding 8,000 words, we employ a two-stage hierarchical approach:

Stage 1: Chunk Summarization The document is divided into overlapping chunks of 5,000 words with 20% overlap (1,000 words). Each chunk is summarized independently with the prompt: "Summarize this legal judgment." This produces intermediate summaries capturing key information from each document section.

Stage 2: Meta-Summarization The chunk summaries are concatenated and processed with a meta-prompt: "Combine these summaries into one coherent summary." This stage synthesizes a final unified summary that maintains coherence throughout the document.

For documents under 8,000 words, direct single-pass summarization is used without chunking.

Design Justification The chunk size of 5,000 words was determined through preliminary experimentation to balance context preservation and computational feasibility. A 20% overlap mitigates boundary effects, ensuring continuity across legal sections that often span multiple paragraphs.

3.4 Training Configuration

We fine-tune LED-large on the training set with the configuration shown in Table 1.

Parameter	Value
Optimizer	Adafactor, learning rate 3e-5
Batch size	1 per device, 16 gradient accumulation steps (effective 16)
Training epochs	15
Input length	9,192 tokens maximum
Output length	400–768 tokens
Warmup	15% of training steps
Weight decay	0.01
Mixed precision	FP16 on GPU
Checkpointing	Every 500 steps, retain last 3

Table 1: Training configuration for fine-tuning LED-large on the InLSum dataset.

Gradient checkpointing is enabled to reduce memory consumption, allowing larger effective batch sizes through gradient accumulation. The final model, after 15 epochs of training, is used for inference on the test set.

3.5 Generation Strategy

The beam search parameters used during inference are summarized in Table 2.

Parameter	Value
Beam size	8
Length penalty	1.2 (encourages longer summaries)
Repetition penalty	1.5
No-repeat n-gram size	4
Early stopping	Enabled

Table 2: Beam search parameters used during inference.

Post-processing removes only strict sentence-level duplicates by normalizing and hashing full sentences. This ensures that legally meaningful repetitions are preserved while eliminating redundant generated text.

4 Experimental Setup

4.1 Dataset

The InLSum dataset[2] consists of Indian legal judgments paired with reference summaries written by experts. Table 3 provides detailed token-level statistics for training, validation, and test splits using the LED tokenizer.

4.2 Evaluation Metrics

Model quality was assessed using ROUGE (Lin, 2004) and BLEU (Papineni et al., 2002). ROUGE-2 and ROUGE-L evaluate overlap and sequence consistency, while BLEU measures n-gram precision and fluency characteristics.

The final ranking is determined by the average of ROUGE-2, ROUGE-L, and BLEU scores. This combined metric balances different aspects of summary quality: bigram overlap (ROUGE-2) captures content selection, sequence matching (ROUGE-L) evaluates structural coherence, and BLEU evaluates overall n-gram precision and fluency.

4.3 Implementation Details

This system is implemented in PyTorch using the HuggingFace Transformers library. Training was

[2] https://exploration-lab.github.io/JUST-NLP/task/

Statistic	Value
Training samples	1200 pairs
Validation samples	200 judgments
Test samples	400 judgments
Judgment length range	229 – 329,814 tokens
Median judgment length	4,360 tokens
Average judgment length	10,926 tokens
Summary length range	39 – 2,689 tokens
Median summary length	672 tokens
Average summary length	708 tokens
Domain	Indian legal court judgments
Tokenizer used	`allenai/led-large-16384`

Table 3: Token-level statistics of the InLSum dataset using the LED tokenizer.

conducted on Nvidia RTX A6000. Key implementation choices include:

- Schema-robust data loading handling column name variations
- Safe iteration over DataFrame rows during inference
- Runtime fallback for out-of-memory cases using truncation
- Comprehensive error handling for edge cases

5 Results and Analysis

5.1 Main Results

Table 4 presents the system's performance on the InLSum test set. The hierarchical LED-based approach achieves competitive results across all metrics, with ROUGE-2 of 29.62, ROUGE-L of 28.56, and BLEU of 21.67, yielding an average score of 26.62. The obtained metrics indicate that the proposed system extracts salient legal content and produces fluent summaries.

Our system achieved second rank on the official L-SUMM shared task leaderboard, placing it among the competitive submissions in the shared task. This demonstrates that hierarchical processing, combined with LED's long-context attention, is an effective strategy for summarizing extremely long legal judgments.

Metric	Score
ROUGE-2	29.62
ROUGE-L	28.56
BLEU	21.67
Average	**26.62**

Table 4: Performance of our system on the InLSum test set. All scores are reported as percentages. The average is computed across ROUGE-2, ROUGE-L, and BLEU.

The ROUGE-2 and ROUGE-L scores indicate strong bigram and sequence-level overlap with reference summaries, suggesting our system captures important factual content and legal reasoning patterns. The BLEU score of 21.67 reflects reasonable n-gram precision. The overall average of 26.62 indicates a balanced performance across various evaluation dimensions.

5.2 Error Analysis

Common failure modes include:

- **Citation handling**: Complex citation chains sometimes lose context
- **Multi-party cases**: Cases with numerous parties occasionally conflate identities
- **Procedural details**: The balance between procedural and substantive content varies

6 Conclusion

We presented a hierarchical LED-based system for legal document summarization that effectively handles extremely long judgments through a combination of chunk-level processing and meta-summarization. Using strategic global attention patterns and carefully tuned hyperparameters, our system achieves strong performance on the L-SUMM shared task. The approach successfully captures the structural and semantic complexity of long legal texts.

In future work, we plan to explore the integration of legal knowledge bases for improved citation handling, incorporate multi-task learning with related legal NLP objectives, and investigate adaptive chunking strategies that align with the discourse structure of judgments. Another promising direction is the design of verdict-aware prompting mechanisms to improve the specificity and interpretability of generated summaries. Overall, our results demonstrate that combining long-context architectures with hierarchical summarization pipelines is a practical and effective solution for legal document summarization.

Limitations

This work has several limitations. The hierarchical design, while effective for managing extremely long documents, introduces additional computational overhead and increases inference latency compared to single-pass models. Despite the use of overlapping chunks, segmentation may still split important contextual information, occasionally affecting the continuity of legal reasoning in the final summary. Our system was trained and evaluated exclusively on Indian legal judgments, raising questions about generalizability to other jurisdictions or legal writing styles. Memory constraints restricted batch sizes during training, limiting the extent of hyperparameter exploration and ablation studies. Finally, the meta-summarization stage can sometimes compress information too aggressively, resulting in minor coherence issues in cases involving dense reasoning or complex procedural histories.

References

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Paheli Bhattacharya, Shounak Paul, Kripabandhu Ghosh, Saptarshi Ghosh, and Adam Wyner. 2019. Identification of rhetorical roles of sentences in indian legal judgments. In *Legal knowledge and information systems*, pages 3–12. IOS Press.

Paheli Bhattacharya, Soham Poddar, Koustav Rudra, Kripabandhu Ghosh, and Saptarshi Ghosh. 2021. Incorporating domain knowledge for extractive summarization of legal case documents. In *Proceedings of the eighteenth international conference on artificial intelligence and law*, pages 22–31.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. Legal-bert: The muppets straight out of law school. *arXiv preprint arXiv:2010.02559*.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. *arXiv preprint arXiv:1804.05685*.

Ambedkar Kanapala, Sukomal Pal, and Rajendra Pamula. 2019. Text summarization from legal documents: a survey. *Artificial Intelligence Review*, 51(3):371–402.

Yoshinobu Kano, Mi-Young Kim, Masaharu Yoshioka, Yao Lu, Juliano Rabelo, Naoki Kiyota, Randy Goebel, and Ken Satoh. 2018. Coliee-2018: Evaluation of the competition on legal information extraction and entailment. In *JSAI International Symposium on Artificial Intelligence*, pages 177–192. Springer.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Vedant Parikh, Vidit Mathur, Parth Mehta, Namita Mittal, and Prasenjit Majumder. 2021. Lawsum: A weakly supervised approach for indian legal document summarization. *arXiv preprint arXiv:2110.01188*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and 1 others. 2020. Big bird: Transformers for longer sequences. *Advances in neural information processing systems*, 33:17283–17297.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International conference on machine learning*, pages 11328–11339. PMLR.

Author Index